KAMPUS AKADEMIK PUBLISING
Jurnal Sains Student Research (JSSR)
Vol.4, No.4 Agustus 2026
e-ISSN: 3025-9851, p-ISSN : 3025-986X, Hal 1070-1084
DOI : https://doi.org/10.61722/jssr.v4i4.11545

ANALISIS PERBANDINGAN ALGORITMA RANDOM FOREST DAN ISOLATION FOREST DALAM DETEKSI ANCAMAN KEAMANAN SIBER

Muhamad Farhan Qolbi[1*], Dede Brahma Arianto[2]
[1] Universitas Faletehan
[2] Universitas Faletehan
*Penulis Korespondensi: muhamadfarhanqolbi@gmail.com[1], dedebrahma@uf.ac.id[2]

Abstract. *Cybersecurity threats continue to increase in both number and complexity, necessitating a detection system capable of accurately and efficiently identifying malicious activity. This study aims to evaluate the performance of the Random Forest algorithm compared to the Isolation Forest in detecting cybersecurity threats by utilizing the Cybersecurity Threat Detection Logs Dataset. Experiments were conducted with three variations of data size: 100,000, 200,000, and 500,000 to assess the impact of data scale on model performance. The preprocessing process included feature selection, categorical data encoding, and the division of training and testing data using a stratified sampling technique. Model evaluation was performed using precision, recall, F1-score, and accuracy metrics based on a weighted average due to class imbalance in the dataset. The study findings revealed that Random Forest provided more stable and superior performance with precision, recall, F1-score, and accuracy values reaching 0.85 in all test scenarios. On the other hand, Isolation Forest showed good effectiveness in detecting anomalies, but also produced a higher false positive rate. Therefore, Random Forest is more suitable to be used as the main model for labeled cyber threat detection, while Isolation Forest can serve as a supporting system for anomaly detection.*

Keywords : *Cybersecurity, Threat Detection, Random Forest, Isolation Forest, Machine*

Abstrak. Ancaman terhadap keamanan siber terus menunjukkan peningkatan baik dalam jumlah maupun kompleksitasnya, sehingga diperlukan sistem deteksi yang mampu secara tepat dan efisien mengidentifikasi aktivitas yang berbahaya. Penelitian ini bertujuan untuk mengevaluasi performa algoritma Random Forest dibandingkan dengan Isolation Forest dalam mendeteksi ancaman keamanan siber dengan memanfaatkan Dataset Cybersecurity Threat Detection Logs. Eksperimen dilaksanakan dengan tiga variasi jumlah data, yaitu 100.000, 200.000, dan 500.000 data untuk mengkaji dampak skala data terhadap kinerja model. Proses preprocessing meliputi pemilihan fitur, pengkodean data kategorikal, serta pembagian data pelatihan dan data pengujian menggunakan teknik stratified sampling. Penilaian model dilakukan dengan menggunakan metrik precision, recall, F1-score, dan akurasi berdasarkan weighted average karena adanya ketidakseimbangan kelas pada Dataset. Temuan penelitian mengungkapkan bahwa Random Forest memberikan performa yang lebih stabil dan unggul dengan nilai precision, recall, F1-score, dan akurasi mencapai 0,85 di semua skenario pengujian. Di sisi lain, Isolation Forest menunjukkan efektivitas yang baik dalam mendeteksi anomali, tetapi juga menghasilkan tingkat false positive yang lebih tinggi. Oleh karena itu, Random Forest lebih cocok digunakan sebagai model utama untuk deteksi ancaman siber berlabel, sementara Isolation Forest dapat berfungsi sebagai sistem pendukung untuk deteksi anomali.

Kata Kunci : *Cybersecurity, Deteksi Ancaman, Random Forest, Isolation Forest, Machine Learning*

1. LATAR BELAKANG

Teknologi informasi berperan krusial dalam mendukung keberhasilan bisnis di era digital saat ini. Data menjadi salah satu aset penting bagi perusahaan, lembaga, atau

Naskah Masuk: 28 Juni 2026; Revisi: 28 Juni 2026; Diterima: 29 Juni 2026; ; Terbit: 30 Juni 2026.

institusi pendidikan, baik yang bersifat swasta maupun pemerintah, yang dapat memastikan kelangsungan organisasi melalui pertukaran data yang cepat dan efisien, meskipun dengan mengorbankan aspek keamanan informasi. Dengan semakin majunya jaringan komputer dalam berbagi data, muncul berbagai kerentanan pada keamanan informasi, bahkan dengan tingkat risiko yang tinggi, sebab pihak-pihak tertentu dapat dengan mudah mengakses atau mengungkapkan data orang lain tanpa hak akses. Di samping terdapat ancaman dari luar (eksternal), risiko juga dapat muncul dari dalam (internal), terutama dari karyawan atau personal yang melakukan tindakan curang akibat lemahnya pengawasan terhadap prosedur keamanan informasi yang diberlakukan (Saputra dkk., 2023).

Kejahatan siber mampu menyerang jaringan komputer, menyusup ke dalam jaringan untuk mengambil informasi rahasia dan menonaktifkan sistem komputer. Untuk mengatasi potensi kejahatan yang mungkin muncul, diperlukan sistem yang dilengkapi dengan firewall serta Intrusion Detection System (IDS). Firewall dan IDS berfungsi sebagai komponen keamanan jaringan yang bisa melindungi server, jaringan, serta mencegah serangan (Yuliana dkk., 2022).

Berdasarkan Dataset Cybersecurity Threat Detection Logs yang menjadi sumber data dalam studi ini dan diambil dari open data, peneliti memilih algoritma random forest dan isolation forest untuk menganalisis perbandingan akurasi model. Random Forest merupakan penyempurnaan dari metode Decision Tree yang melibatkan beberapa Decision Tree, di mana setiap Decision Tree dilatih dengan menggunakan sampel terpisah dan setiap atribut dibagi pada pohon berdasarkan subset atribut yang bersifat acak. (Supriyadi dkk., 2020). Random forest menawarkan berbagai manfaat, seperti meningkatkan ketepatan ketika ada data yang tidak lengkap serta tahan terhadap nilai ekstrim, dan juga efisien dalam hal penyimpanan data. Selain itu, Random Forest dilengkapi dengan proses pemilihan fitur yang dapat memilih fitur terbaik, sehingga memperbaiki kinerja model klasifikasi. Dengan adanya pemilihan fitur, jelas bahwa Random Forest dapat beroperasi pada data besar dengan parameter yang rumit secara efisien (Devella dkk., 2020).

Berdasarkan kajian yang dikemukakan oleh Fei Tony, Kai Ming Ting, dan Zhi-Hua Zhou (Tony Liu dkk., 2008). Isolation tree merupakan pendekatan dalam deteksi anomali yang berfokus pada proses isolasi data, bukan pada pemodelan data normal.

Metode ini memanfaatkan karakteristik dasar anomali yang cenderung berjumlah sedikit dan memiliki nilai atribut yang berbeda dibandingkan data normal, sehingga memungkinkan anomali terisolasi lebih cepat dalam struktur pohon. Algoritma ini akan membangun sekumpulan isolation tree. Dalam setiap tree, data akan dipartisi secara rekursif dengan memilih fitur dan nilai pemisah secara acak hingga setiap titik data terisolasi. Anomali, yang berada jauh dari klaster data normal, cenderung akan terisolasi dengan jumlah partisi yang lebih sedikit dan lebih dekat ke akar pohon. Keunggulan utama dari isolation Forest adalah efisiensi komputasi yang tinggi dan skalabilitas yang baik, bahkan pada Dataset besar, karena penggunaan sub sampling data (Pramudita dkk., 2025).

Penelitian berjudul mendeteksi anomali pada data transaksi jual beli akta tanah dengan menggunakan algoritma decision tree, isolation forest, dan random forest yang di buat oleh Yonathan Hartoko Dwi Ari Putro, 2022 (2022) yang hasil pembahasannya menunjukkan bahwa proses pendeteksian anomali dapat dilakukan secara efektif dengan memanfaatkan algoritma machine learning berbasis tree yaitu decision tree, isolation forest, dan random forest. Dataset yang digunakan merupakan data transaksi akta tanah tahun 2018 yang telah melalui tahapan preprocessing, normalisasi, serta identifikasi atribut anomali berdasarkan nilai akta per meter menggunakan pendekatan statistik kuartil. Dari total data yang dianalisis, diperoleh 91 data teridentifikasi sebagai anomali dan 265 data sebagai data normal.

Penelitian berjudul penerapan algoritma random forest untuk deteksi phishing pada website yang di buat oleh Muhammad Fahri, 2025 (2025) hasil pembahasannya menunjukkan bahwa penerapan algoritma random forest mampu memberikan performa yang sangat tinggi dalam mendeteksi website phishing, ditunjukkan dengan capaian akurasi sebesar 98,20%, presisi 98,22%, dan F1-score 98,22%, yang menandakan keseimbangan yang baik antara kemampuan mendeteksi website phishing dan meminimalkan kesalahan klasifikasi.

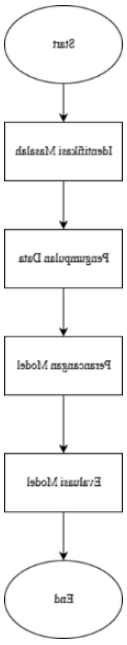
Penelitian berjudul deteksi trafik anomali berdasarkan pola trafik menggunakan isolation forest yang dibuat oleh Muhammad ‘ Azam Al-Akbar, Ardan Pratama Y, Agil Naufal Al Habib Gurning, 2025 (2025) hasil pembahasannya menunjukkan bahwa

algoritma isolation forest mampu mendeteksi trafik jaringan yang menyimpang secara statistik dengan cukup baik menggunakan pendekatan unsupervised learning pada Dataset LUFlow. Melalui tahapan preprocessing, encoding, standarisasi fitur, dan sampling data, model berhasil mengisolasi sekitar 1% trafik sebagai anomali yang di tandai oleh nilai ekstrem pada fitur seperti bytes_in, bytes_out, dan entropy.

Penelitian ini bertujuan untuk membandingkan kinerja dua algoritma machine learning yaitu random forest dan isolation forest dalam mendeteksi ancaman keamanan siber berdasarkan Dataset Cybersecurity Threat Detection Logs. penelitian ini di fokuskan untuk mengetahui perbedaan performa kedua metode tersebut dalam mengidentifikasi pola ancaman, baik dari sisi akurasi maupun kemampuan deteksi terhadap data yang bersifat normal dan anomali. Dengan membandingkan kedua algoritma pada Dataset Cybersecurity Threat Detection Logs, penelitian ini diharapkan dapat memberikan gambaran algoritma mana yang lebih efektif dan sesuai untuk di terapkan dalam sistem deteksi ancaman jaringan khususnya pada lingkungan dengan karakteristik data log yang besar, kompleks, dan berpotensi tidak seimbang.

2. METODE PENELITIAN

Metodologi Penelitian



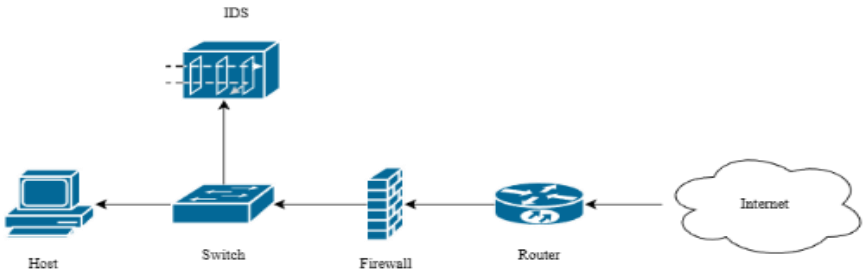
Gambar 1. Metode Penelitian

Proses penelitian ini digambarkan dalam bentuk flowchart sebagaimana ditunjukkan pada Gambar 1. Flowchart ini mempresentasikan alur utama perbandingan

random forest dan isolation forest dalam deteksi ancaman pada Dataset Cybersecurity Threat Detection Logs yang terdiri dari beberapa tahapan: Identifikasi Masalah, Pengumpulan Data, Perancangan Model, Evaluasi Model.

Identifikasi Masalah

Identifikasi masalah pada penelitian ini adalah meningkatnya ancaman keamanan siber yang terekam dalam bentuk Dataset Cybersecurity Threat Detection Logs dengan karakteristik data yang besar, kompleks, dan tidak seimbang antara data normal dan data ancaman. Kondisi tersebut menyulitkan proses identifikasi ancaman secara manual serta berpotensi menurunkan efektivitas sistem keamanan jaringan. Selain itu, belum adanya kejelasan mengenai algoritma machine learning yang paling efektif untuk mendeteksi ancaman pada data log keamanan khususnya antara random forest yang bersifat supervised learning dan isolation forest yang bersifat unsupervised learning. Perbedaan pendekatan kedua algoritma tersebut menimbulkan pertanyaan terkait kemampuan masing – masing model dalam mengenali pola ancaman baik yang bersifat jelas maupun tersembunyi. Oleh karena itu, diperlukan suatu analisis dan perbandingan metode untuk mengetahui algoritma mana yang lebih optimal dalam mendeteksi ancaman keamanan siber berdasarkan Dataset Cybersecurity Threat Detection Logs.



Gambar 2 Analogi Deteksi Ancaman Siber

Pengumpulan Data

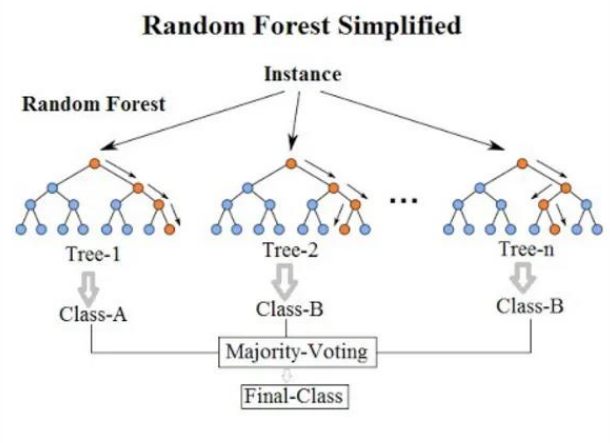
Pengumpulan data pada penelitian ini dilakukan dengan menggunakan Dataset Cybersecurity Threat Detection Logs yang diperoleh dari open data. Dataset ini berisi kumpulan data log keamanan siber yang merepresentasikan aktivitas jaringan baik yang bersifat normal maupun yang mengindikasikan adanya ancaman keamanan. Pemilihan Dataset ini didasarkan pada karakteristik datanya yang realistis, berskala besar, serta

umum digunakan dalam penelitian di bidan keamanan siber dan machine learning sehingga relevan untuk menguji performa algoritma random forest dan isolation forest. Selain itu, Dataset ini dipilih karena telah melalui proses kurasi dan dapat diakses secara terbuka sehingga mendukung transparasi serta replikasi penelitian. Data yang diperoleh kemudian digunakan sebagai input utama dalam proses pemodelan dan evaluasi deteksi ancaman keamanan siber.

Perancangan Model

Pada fase perancangan model Dataset Cybersecurity Threat Detection Logs yang diambil dari data terbuka ini dimanfaatkan untuk membuat dua model deteksi ancaman yaitu random forest dan isolation forest. Pemilihan kedua algoritma ini didasarkan pada karakteristik unik yang dimiliki masing-masing dalam proses pembelajaran dan identifikasi pola ancaman pada data log jaringan.

Random Forest adalah algoritma pembelajaran terawasi yang menggunakan pendekatan ensembel dengan membangun beberapa pohon keputusan dan mengkombinasikan hasil prediksi dari setiap pohon untuk menentukan kelas akhir. Strategi ini memungkinkan model untuk mempelajari pola ancaman secara sistematis berdasarkan data yang sudah dilabeli serta meningkatkan ketepatan dan kestabilan prediksi. Dalam penelitian ini, Random Forest diterapkan untuk mengklasifikasikan aktivitas jaringan ke dalam kategori normal dan ancaman berdasarkan fitur-fitur yang terdapat dalam dataset. Gambar 3 memberikan visualisasi dari algoritma Random Forest.



Gambar 3 Random Forest

Untuk rumus algoritma Random Forest sebagai berikut:

Entropy

$$Entropy\ (Y) = -\sum_i p\ (c|Y) log^2 p(c|Y), \quad (1)$$

Keterangan :

Y = Himpunan kasus

P(c|Y) = Proporsi nilai Y terhadap kelas c.

Information Gain

$$(Y, a) = Entropy\ (Y) - \quad (2)$$

$$\sum_v \varepsilon\, Values(a) \frac{|Y_v|}{|Y_a|}\ Entropy\ (Y_v).$$

Keterangan :

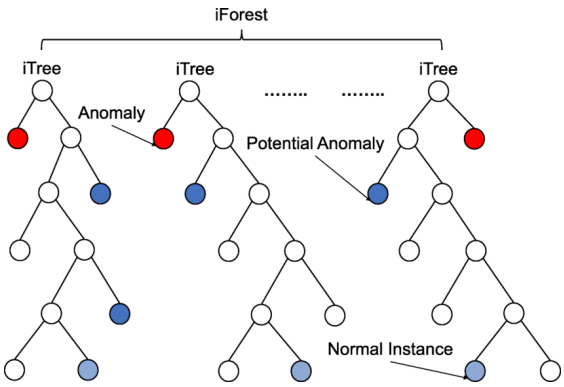
$Values(a)$ = Nilai yang mungkin dalam himpunan kasus a

Y_v = Subkelas dari Y dengan kelas v yang berhubungan dengan kelas a

Y_a = Semua nilai yang sesuai dengan a

Isolation Forest merupakan algoritma *unsupervised learning* yang dirancang khusus untuk mendeteksi anomaly dengan cara mengisolasi data yang memiliki karakteristik berbeda dari mayoritas data. Algoritma ini bekerja dengan membangun *Isolation Tree* secara acak di mana data yang bersifat anomaly cenderung terisolasi lebih cepat dibandingkan data normal. Dalam penelitian ini, Isolation Forest digunakan untuk mengidentifikasi aktivitas jaringan yang menyimpang dari pola normal sebagai indikasi potensi ancaman keamanan siber.

Pada Gambar 4 itu merupakan Gambaran dari algoritma Isolation Forest.



Gambar 4 Isolation Forest

Untuk rumus algoritma Isolation Forest sebagai berikut :

Skor Anomali

$$s(x,n) = 2 - \frac{E(h(x))}{c(n)} \quad (3)$$

Keterangan :

E(h(x)) = Rata – rata panjang jalur data x

n = jumlah data

c(n) = normalisasi panjang jalur

Panjang Jalur Rata – Rata (Average Path Length)

$$c(n) = 2H(n-1) - \frac{2(n-1)}{n} \quad (4)$$

Evaluasi Model

Evaluasi model dilaksanakan untuk menilai performa dari algoritma Random Forest dan Isolation Forest dalam mengenali ancaman di bidang keamanan siber berdasarkan Dataset Log Deteksi Ancaman Siber. Ukuran performa model dilakukan dengan menerapkan beragam metrik evaluasi seperti *precision*, *recall*, *F1-score*, dan *accuracy* yang dihasilkan dari confusion matrix.

Precision berfungsi untuk menilai seberapa akurat model dalam mengidentifikasi data sebagai ancaman, yaitu proporsi data yang benar-benar merupakan ancaman dari keseluruhan data yang dikategorikan sebagai ancaman. Metrik ini sangat penting untuk mengevaluasi kemampuan model dalam mengurangi kesalahan *false positive*.

Recall diukur untuk menentukan sejauh mana model dapat menemukan semua data ancaman yang ada di dalam dataset. Tingkat recall yang tinggi menunjukkan bahwa model berhasil mendeteksi sebagian besar aktivitas yang sebenarnya merupakan ancaman.

F1-score adalah rata-rata harmonis antara *precision* dan *recall* yang memberikan gambaran mengenai keseimbangan kinerja model, terutama dalam situasi di mana distribusi kelas dalam Dataset tidak seimbang.

Accuracy digunakan untuk menilai seberapa akurat model dalam mengklasifikasikan semua data, baik yang normal maupun yang dianggap sebagai

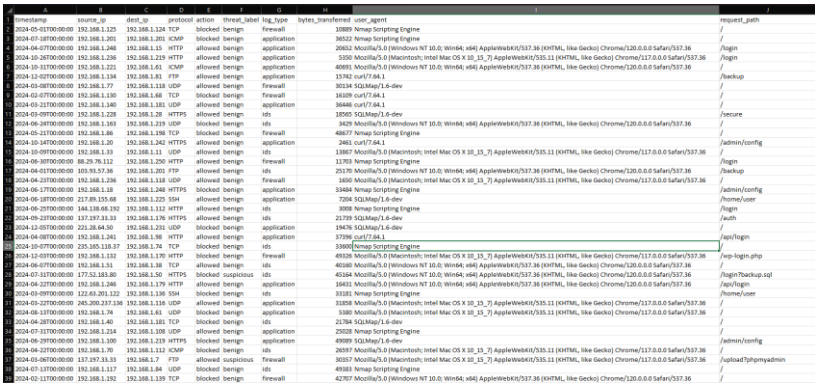
ancaman. Metrik ini menunjukkan persentase prediksi yang benar dibandingkan dengan seluruh data yang diuji.

Dalam penelitian ini nilai *precision*, *recall*, *F1-score*, dan *accuracy* dihitung dengan pendekatan *weighted average* untuk mempertimbangkan ketidakseimbangan kelas antara data normal dan data ancaman. Hasil evaluasi ini kemudian digunakan sebagai landasan untuk membandingkan performa antara algoritma Random Forest yang termasuk dalam *supervised learning* dan Isolation Forest yang termasuk dalam *unsupervised learning*.

3. HASIL DAN PEMBAHASAN

Pengumpulan dan Penentuan Jumlah Data

Dataset yang digunakan dalam penelitian ini adalah Cybersecurity Threat Detection Logs yang diperoleh dari open data. Dataset ini berisi data log aktivitas jaringan yang mencerminkan berbagai kondisi normal maupun aktivitas yang mengindikasikan adanya ancaman keamanan siber. Dari kesuluruhan Dataset, peneliti ini menggunakan tiga sampel data yaitu 100.000, 200.000, 500.000 data yang diambil secara acak dengan parameter random_state untuk menjaga konsistensi hasil eksperimen. Pemilihan jumlah data ini bertujuan untuk mempresentasikan karakteristik data log keamanan yang berskala besar serta menguji kinerja model pada kondisi yang mendekati penerapan di lingkungan nyata.



	A	B	C	D	E	F	G	H	I	J
1	timestamp	source_ip	dest_ip	protocol	action	threat_label	log_type	bytes_transferred	user_agent	request_path
2	2024-05-01T00:00:00	192.168.1.125	192.168.1.124	TCP	blocked	benign	firewall	10889	Nmap Scripting Engine	/
3	2024-07-18T00:00:00	192.168.1.201	192.168.1.201	ICMP	blocked	benign	application	36522	Nmap Scripting Engine	/
4	2024-04-07T00:00:00	192.168.1.248	192.168.1.15	HTTP	allowed	benign	application	20652	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/120.0.0.0 Safari/537.36	/login
5	2024-10-26T00:00:00	192.168.1.236	192.168.1.219	HTTP	allowed	benign	application	5350	Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/535.11 (KHTML, like Gecko) Chrome/117.0.0.0 Safari/537.36	/login
6	2024-10-31T00:00:00	192.168.1.221	192.168.1.61	ICMP	allowed	benign	application	40691	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/120.0.0.0 Safari/537.36	/
7	2024-12-02T00:00:00	192.168.1.134	192.168.1.81	FTP	allowed	benign	application	15762	curl/7.64.1	/backup
8	2024-03-08T00:00:00	192.168.1.77	192.168.1.118	UDP	allowed	benign	firewall	30134	SQLMap/1.6-dev	/
9	2024-02-07T00:00:00	192.168.1.130	192.168.1.68	TCP	blocked	benign	firewall	16109	curl/7.64.1	/
10	2024-03-21T00:00:00	192.168.1.140	192.168.1.181	UDP	allowed	benign	application	36446	curl/7.64.1	/
11	2024-03-09T00:00:00	192.168.1.228	192.168.1.28	HTTPS	allowed	benign	ids	18565	SQLMap/1.6-dev	/secure
12	2024-08-24T00:00:00	192.168.1.163	192.168.1.219	UDP	blocked	benign	ids	3429	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/120.0.0.0 Safari/537.36	/
13	2024-05-21T00:00:00	192.168.1.86	192.168.1.198	TCP	blocked	benign	firewall	48677	Nmap Scripting Engine	/
14	2024-10-14T00:00:00	192.168.1.20	192.168.1.242	HTTPS	allowed	benign	application	2461	curl/7.64.1	/admin/config
15	2024-10-09T00:00:00	192.168.1.33	192.168.1.11	UDP	allowed	benign	ids	13867	Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/535.11 (KHTML, like Gecko) Chrome/117.0.0.0 Safari/537.36	/
16	2024-06-30T00:00:00	88.29.76.112	192.168.1.250	HTTP	allowed	benign	firewall	11703	Nmap Scripting Engine	/login
17	2024-04-01T00:00:00	103.93.57.36	192.168.1.201	FTP	allowed	benign	ids	25170	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/120.0.0.0 Safari/537.36	/backup
18	2024-04-23T00:00:00	192.168.1.236	192.168.1.118	UDP	allowed	benign	firewall	1650	Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/535.11 (KHTML, like Gecko) Chrome/117.0.0.0 Safari/537.36	/
19	2024-06-17T00:00:00	192.168.1.18	192.168.1.248	HTTPS	blocked	benign	application	33484	Nmap Scripting Engine	/admin/config
20	2024-06-18T00:00:00	217.89.155.68	192.168.1.225	SSH	allowed	benign	application	7204	SQLMap/1.6-dev	/home/user
21	2024-06-25T00:00:00	144.138.68.192	192.168.1.112	HTTP	allowed	benign	ids	3008	Nmap Scripting Engine	/login
22	2024-09-23T00:00:00	137.197.33.33	192.168.1.176	HTTPS	allowed	benign	ids	21739	SQLMap/1.6-dev	/auth
23	2024-12-05T00:00:00	221.28.64.50	192.168.1.231	UDP	blocked	benign	application	19476	SQLMap/1.6-dev	/
24	2024-04-08T00:00:00	192.168.1.241	192.168.1.96	HTTP	allowed	benign	application	37396	curl/7.64.1	/api/login
25	2024-10-07T00:00:00	255.185.118.37	192.168.1.74	TCP	blocked	benign	ids	33001	Nmap Scripting Engine	/
26	2024-12-03T00:00:00	192.168.1.132	192.168.1.170	HTTP	blocked	benign	firewall	49328	Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/535.11 (KHTML, like Gecko) Chrome/117.0.0.0 Safari/537.36	/wp-login.php
27	2024-06-07T00:00:00	192.168.1.51	192.168.1.38	TCP	allowed	benign	ids	40160	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/120.0.0.0 Safari/537.36	/
28	2024-07-31T00:00:00	177.52.183.80	192.168.1.50	HTTPS	blocked	suspicious	ids	45164	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/120.0.0.0 Safari/537.36	/login?backup.sql
29	2024-04-22T00:00:00	192.168.1.246	192.168.1.179	HTTP	allowed	benign	application	16431	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/120.0.0.0 Safari/537.36	/api/login
30	2024-03-09T00:00:00	122.63.201.122	192.168.1.136	SSH	blocked	benign	ids	33181	Nmap Scripting Engine	/home/user
31	2024-03-22T00:00:00	245.200.237.136	192.168.1.116	UDP	allowed	benign	application	31858	Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/535.11 (KHTML, like Gecko) Chrome/117.0.0.0 Safari/537.36	/
32	2024-08-13T00:00:00	192.168.1.74	192.168.1.61	UDP	blocked	benign	application	5380	Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/535.11 (KHTML, like Gecko) Chrome/117.0.0.0 Safari/537.36	/
33	2024-04-28T00:00:00	192.168.1.40	192.168.1.181	TCP	blocked	benign	ids	21784	SQLMap/1.6-dev	/
34	2024-07-31T00:00:00	192.168.1.214	192.168.1.108	UDP	allowed	benign	application	25028	Nmap Scripting Engine	/
35	2024-06-29T00:00:00	192.168.1.100	192.168.1.219	HTTPS	blocked	benign	application	49089	SQLMap/1.6-dev	/admin/config
36	2024-04-22T00:00:00	192.168.1.70	192.168.1.112	ICMP	blocked	benign	ids	26597	Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/535.11 (KHTML, like Gecko) Chrome/117.0.0.0 Safari/537.36	/
37	2024-03-06T00:00:00	127.197.33.33	192.168.1.7	FTP	allowed	suspicious	firewall	30357	Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/535.11 (KHTML, like Gecko) Chrome/117.0.0.0 Safari/537.36	/upload?phpmyadmin
38	2024-07-13T00:00:00	192.168.1.117	192.168.1.84	UDP	blocked	benign	ids	49383	Nmap Scripting Engine	/
39	2024-02-11T00:00:00	192.168.1.192	192.168.1.139	TCP	blocked	benign	firewall	42707	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/120.0.0.0 Safari/537.36	/

Gambar 5 ***Dataset Cybersecurity Threat Detection Log***

Preprocessing **Data dan** ***Encoding*** **Label**

Pada tahap preprocessing data untuk meningkatkan kualitas dan relevansi fitur yang digunakan, beberapa atribut yang tidak berkontribusi langsung terhadap proses deteksi ancaman, yaitu timestamp, sourceIP, destinationIP, request path, dan user agent dihapus dari Dataset. Selanjutnya atribut kategorikal seperti protocol, action, log type, dan threat label diubah ke dalam bentuk numerik menggunakan metode Label Encoding untuk menjaga interpretabilitas hasil, proses encoding ini juga disertai dengan penyimpanan mapping label ke dalam berkas terpisah sehingga hasil prediksi model dapat dikonversi kembali ke nama label ancaman aslinya dan memudahkan hasil analisis penelitian.

Dataset yang telah diproses kemudian dibagi menjadi data latih dan data uji dengan rasio 80:20 menggunakan metode stratified sampling. Teknik ini digunakan untuk menjaga proporsi distribusi kelas antara data latih dan data uji tetap seimbang mengingat Dataset memiliki karakteristik tidak seimbang antara data normal dan data ancaman. Data latih digunakan untuk membangun model random forest dan isolation forest sedangkan data uji digunakan untuk mengevaluasi kinerja kedua model tersebut.

Berikut script untuk model Random Forest :

```
rf = RandomForestClassifier(n_estimators=200, class_weight='balanced',
    max_depth=None, n_jobs=2, random_state=42)
rf.fit(X_train, y_train)
y_pred = rf.predict(X_test)
import json
with open("label_mappings.json", "r") as f:
  label_mappings = json.load(f)
inv_map = {int(v): k for k, v in label_mappings['threat_label'].items()}
def get_label_name(val):
  return inv_map.get(int(val), str(val))
rf_y_test_named = [get_label_name(label) for label in y_test]
rf_y_pred_named = [get_label_name(label) for label in y_pred]

print("=== Random Forest Results ===")
print("Accuracy:", accuracy_score(rf_y_test_named, rf_y_pred_named))
print(classification_report(rf_y_test_named, rf_y_pred_named))
```

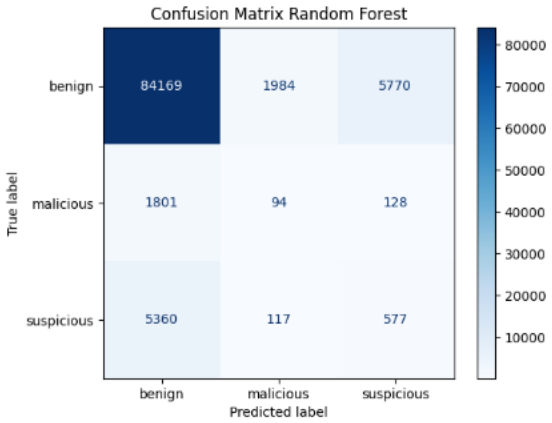
Berikut script untuk model Isolation Forest :

```
y_test_binary = (y_test != 0).astype(int)
iso = IsolationForest(n_estimators=100, max_samples=500000, n_jobs=2,
        contamination=0.1, random_state=42)
iso.fit(X_train)
iso_pred_raw = iso.predict(X_test)
iso_pred = [1 if x == -1 else 0 for x in iso_pred_raw]
```

```
import json
with open("label_mappings.json", "r") as f:
    label_mappings = json.load(f)
inv_map = {int(v): k for k, v in label_mappings['threat_label'].items()}
def get_label_name(val):
    return inv_map.get(int(val), str(val))
y_test_named = [get_label_name(label) for label in y_test]
iso_pred_named = [get_label_name(label) for label in iso_pred]

print("\n=== Isolation Forest Results ===")
print(classification_report(y_test_named, iso_pred_named))
```

```
=== Random Forest Results ===
Accuracy: 0.8484
              precision    recall  f1-score   support

      benign       0.92      0.92      0.92     91923
   malicious       0.04      0.05      0.04      2023
  suspicious       0.09      0.10      0.09      6054

    accuracy                           0.85    100000
   macro avg       0.35      0.35      0.35    100000
weighted avg       0.85      0.85      0.85    100000
```

Gambar 6 Hasil Evaluasi Random Forest



Gambar 7 *Congfusion Matrix Random Forest*

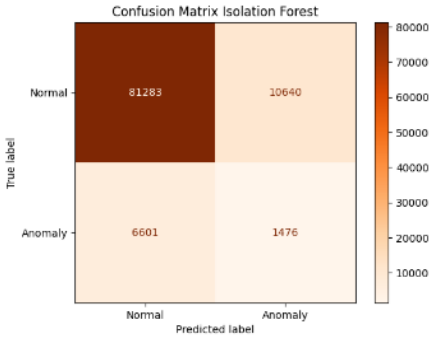
Hasil Evaluasi Isolation Forest

Pada hasil model isolation forest yang digunakan sebagai metode deteksi anomali dengan pendekatan unsupervised learning. Pada tahap evaluasi, label ancaman dikonversi ke dalam bentuk biner, di mana data normal direpresentasikan dengan nilai nol dan data anomali direpresentasikan dengan nilai satu. Model isolation forest dilatih dengan asumsi

bahwa sekitar sepuluh persen data merupakan anomali. Hasil pengujian menunjukkan bahwa isolation forest mampu mendeteksi sebagian besar aktivitas yang menyimpang dari pola normal, namun masih menghasilkan tingkat false positive yang cukup tinggi di mana beberapa data normal terdeteksi sebagai anomali. Kondisi ini merupakan karakteristik umum dari metode unsupervised yang tidak memanfaatkan informasi label selama proses pelatihan.

```
=== Isolation Forest Results ===
              precision    recall  f1-score   support

      benign       0.92      0.88      0.90     91923
   malicious       0.04      0.24      0.07      2023
  suspicious       0.00      0.00      0.00      6054

    accuracy                           0.82    100000
   macro avg       0.32      0.38      0.32    100000
weighted avg       0.85      0.82      0.83    100000
```

Gambar 8 Hasil Evaluasi *Isolation Forest*



Gambar 9 *Confusion Matrix Isolation Forest*

Perbandingan Model

Perbandingan antar model dilakukan untuk mengevaluasi performa algoritma random forest dan isolation forest dalam mengidentifikasi ancaman keamanan siber dengan menggunakan sampel sebanyak 100.000, 200.000, dan 500.000 data. Penilaian kinerja model dilakukan dengan memanfaatkan metrik precision, recall, F1-score, dan accuracy yang diambil dari nilai rata-rata tertimbang pada laporan klasifikasi, mengingat ketidakseimbangan distribusi kelas dalam Dataset.

Berdasarkan hasil evaluasi terhadap ketiga sampel data tersebut, algoritma random forest menunjukkan kinerja yang lebih stabil dibandingkan dengan isolation forest. Pada semua variasi sampel data, yaitu 100.000, 200.000, dan 500.000, random forest memperoleh nilai precision, recall, F1-score, dan accuracy yang dapat

dipertahankan di angka 0,85. Konsistensi nilai ini menunjukkan bahwa random forest memiliki kemampuan untuk menggeneralisasi dengan baik dan dapat menjaga performa klasifikasi meskipun jumlah data yang digunakan untuk pelatihan mengalami kenaikan yang signifikan.

Di sisi lain, isolation forest menunjukkan kinerja yang kurang baik jika dibandingkan dengan random forest, terutama dalam metrik recall, F1-score, dan accuracy. Pada ketiga sampel data, isolation forest mencatat nilai accuracy yang berkisar antara 0,81 hingga 0,82, dengan nilai recall yang lebih rendah dibandingkan random forest. Hal ini menunjukkan bahwa isolation forest cukup efektif dalam mendeteksi data anomalus, tetapi model ini mengalami tingkat kesalahan klasifikasi yang lebih tinggi, terutama dalam membedakan antara data normal dan data yang mencurigakan.

Tabel 1 Perbandingan *Algoritma Random Forest dan Isolation Forest*

Nama Model	Jumlah Data	Precision	Recall	F1-Score	Accuracy
Random Forest	100.000	0.85	0.85	0.85	**0.85**
	200.000	0.85	0.85	0.85	**0.85**
	500.000	0.85	0.85	0.85	**0.85**
Isolation Forest	100.000	0.85	0.81	0.83	**0.81**
	200.000	0.85	0.81	0.83	**0.81**
	500.000	0.85	0.82	0.83	**0.82**

Perbedaan kinerja tersebut dipengaruhi oleh karakteristik pendekatan pembelajaran yang digunakan oleh masing – masing algoritma. Random forest sebagai metode supervised learning memanfaatkan informasi label ancaman selama proses pelatihan sehingga mampu mempelajari pola ancaman secara lebih akurat dan terstruktur. Sebaliknya isolation forest menggunakan pendekatan unsupervised learning yang berfokus pada proses isolasi data yang menyimpang dari pola normal tanpa memanfaatkan label kelas sehingga sensitivitas terhadap anomali menjadi lebih tinggi namun di ikuti dengan peningkatan false positive.

Dengan demikian hasil perbandingan ini menunjukkan bahwa random forest lebih sesuai digunakan sebagai model utama dalam sistem deteksi ancaman keaman siber berbasis data log yang telah memiliki label ancaman yang jelas. Sementara itu, isolation forest lebih tepat digunakan sebagai model pendukung untuk mendeteksi aktivitas anomali yang belum terdefinisi secara eksplisit dalam label sehingga dapat berperan

sebagai sistem peringatan dini dalam lingkungan jaringan yang dinamis dan berskala besar.

4. KESIMPULAN DAN SARAN

Kesimpulan

Berdasarkan temuan dari penelitian yang telah dilakukan pada Dataset Log Deteksi Ancaman Keamanan Siber dengan tiga ukuran sampel data yaitu 100.000, 200.000, dan 500.000, dapat disimpulkan bahwa algoritma hutan acak dan hutan isolasi efektif untuk mendeteksi ancaman keamanan digital. Evaluasi yang dilakukan mengindikasikan bahwa hutan acak menunjukkan kinerja yang superior dan konsisten dibandingkan dengan hutan isolasi, dengan nilai precision, recall, F1-score, dan akurasi yang stabil pada angka 0,85 di seluruh ukuran sampel uji. Keunggulan ini diperoleh karena pendekatan pembelajaran terawasi yang memungkinkan hutan acak untuk belajar pola ancaman secara langsung dari dataset yang sudah dilabeli, sehingga menghasilkan kemampuan generalisasi yang baik. Di sisi lain, hutan isolasi menunjukkan performa yang sedikit lebih rendah dengan nilai akurasi berkisar antara 0,81 hingga 0,82, namun tetap efektif dalam mendeteksi aktivitas yang tidak biasa sebagai sistem peringatan awal. Dengan demikian, hutan acak lebih tepat untuk digunakan sebagai model utama dalam sistem deteksi ancaman siber berbasis klasifikasi, sementara hutan isolasi dapat berfungsi sebagai model pendukung untuk mengenali aktivitas mencurigakan yang belum diberikan label.

Saran

Penelitian selanjutnya disarankan untuk mengembangkan metode deteksi ancaman dengan mengombinasikan algoritma Random Forest dan Isolation Forest dalam pendekatan hybrid atau ensemble guna meningkatkan akurasi dan menekan tingkat false point. Selain itu, penggunaan Dataset dengan variasi jenis serangan yang lebih beragam serta penerapan data log secara real-time diharapkan dapat memberikan gambaran kinerja sistem deteksi ancaman yang lebih representatif terhadap kondisi nyata.

Penulis mengucapkan terimakasih kepada Fakultas Sains dan Teknik program studi Informatika Universitas Faletehan terutama bapak dosen pembimbing Dede Brahma

Arianto, S.Kom., M.Kom. atas dukungan dan fasilitas yang diberikan sehingga penelitian ini dapat diselesaikan dengan baik.

DAFTAR REFERENSI

Al-Akbar, M. ’Azam, Pratama, A., & Gurning, A. N. A. H. (2025). *Deteksi Trafik Anomali Berdasarkan Pola Trafik Menggunakan Isolation Forest*. https://journal.aira.or.id/index.php/cosmic/article/view/1188

Ariputro, Y. H. D. (2022). *MENDETEKSI ANOMALI PADA DATA TRANSAKSI JUAL BELI AKTA TANAH DENGAN MENGGUNAKAN ALGORITMA DECISION TREE, ISOLATION FOREST, DAN RANDOM FOREST*. https://lib.mercubuana.ac.id/

Devella, S., Yohannes, & Rahmawati, F. N. (2020). Implementasi Random Forest Untuk Klasifikasi Motif Songket Palembang Berdasarkan SIFT. *Jurnal Teknik Informatika dan Sistem Informasi*, *7*(2). http://jurnal.mdp.ac.id

Fahri, M. (2025). Penerapan Algoritma Random Forest untuk Deteksi Phishing pada Website. *JITSI : Jurnal Ilmiah Teknologi Sistem Informasi*, *6*(2), 186–194. https://doi.org/10.62527/jitsi.6.2.472

Pramudita, R. Y., Setiawan, B. D., & Data, M. (2025). *Kinerja Deteksi Tautan Website Phishing Menggunakan Isolation Forest* (Vol. 9, Nomor 10). http://j-ptiik.ub.ac.id

Saputra, L. A., Akbar, F. M., Cahyaningtias, F., Ningrum, M. P., & Fauzi, A. (2023). *Ancaman Keamanan Pada Sistem Informasi Manajemen Perusahaan*. https://doi.org/10.38035/jpsn.v1i12

Supriyadi, R., Gata, W., Maulidah, N., & Fauzi, A. (2020). *Penerapan Algoritma Random Forest Untuk Menentukan Kualitas Anggur Merah*. *13*(2), 67–75. http://journal.stekom.ac.id/index.php/E-Bisnis■page67

Tony Liu, F., Ming Ting, K., & Zhou, Z.-H. (2008). *Isolation Forest*. https://www.researchgate.net/publication/224384174_Isolation_Forest

Yuliana, Y., Mooduto, H. A., & Hadi, R. (2022). Deteksi Ancaman Keamanan Pada Server dan Jaringan Menggunakan OSSEC. Dalam *Jurnal Ilmiah Teknologi Sistem Informasi* (Vol. 3, Nomor 1). http://jurnal-itsi.org