Analisis Performa Isolation Forest dan Extended Isolation Forest untuk Deteksi Anomali pada Trafik QUIC

1st Ruben William Bakara
Program Studi Rekayasa Keamanan Siber
Politeknik Negeri Batam
Batam, Indonesia
ruben.4332201033@students.polibatam.ac.id

2st Hamdani Arif
Program Studi Rekayasa Keamanan Siber
Politeknik Negeri Batam
Batam, Indonesia
hamdaniarif@polibatam.ac.id

***Abstract*— Riset ini menghasilkan performa Isolation Forest (IF) dibandingkan Extended Isolation Forest (EIF) dalam konteks deteksi anomali tanpa label (unsupervised) pada trafik protokol QUIC. Lima puluh satu fitur numerik dari trafik tersebut diuji pada sepuluh tingkat proporsi anomali (1% sampai 10%), baik dalam kondisi data mentah maupun setelah ditransformasi dengan Power Transformer metode Yeo-Johnson. Empat metrik dipakai untuk menilai performa precision, recall, F1-score, AUPRC ditambah pencatatan waktu eksekusi dan konsumsi memori. Ketika diuji pada data mentah, IF mencatat rata-rata F1-score 0,424, jauh melampaui EIF yang hanya 0,103, dengan keunggulan ini konsisten di seluruh sepuluh proporsi berdasarkan 20 kali pengulangan seed acak dan uji statistik Wilcoxon signed-rank (p<0,001). Begitu Power Transformer diterapkan, rata-rata F1-score keduanya naik IF ke 0,603 dan EIF ke 0,622 namun selisih ini tidak lagi signifikan secara statistik pada seluruh proporsi (p>0,05). Artinya, transformasi tersebut tidak membuat EIF unggul, melainkan hanya menyetarakan performanya dengan IF, sementara dari sisi efisiensi memori EIF tetap kalah karena kebutuhannya lebih besar dibanding IF.**

***Keywords*—** *trafik QUIC, unsupervised anomaly detection, Isolation Forest, Extended Isolation Forest, transformasi Power Transformer, optimasi ambang batas (threshold)*

I. PENDAHULUAN

QUIC (Quick UDP Internet Connections) adalah protokol transport generasi baru berbasis UDP yang distandardisasi IETF pada 2021 sebagai fondasi HTTP/3, dengan keamanan terintegrasi melalui TLS 1.3 secara default [1]. Karena hampir seluruh muatan paket dan sebagian header transportnya terenkripsi, sehingga dibutuhkan metode yang mengenali penyimpangan pola dari karakteristik statistik flow. Karena tidak membutuhkan data berlabel, pendekatan unsupervised menjadi pilihan yang relevan untuk kasus ini. Cara kerja Isolation Forest adalah mengisolasi titik data secara acak berdasarkan asumsi bahwa titik yang menyimpang dari pola umum akan lebih cepat terpisah dari titik lain (tercermin dari path length yang lebih pendek) kompleksitas komputasinya yang linear menjadikannya cocok diterapkan pada volume data besar [2]. Meski demikian, IF punya kelemahan struktural karena arah pemisahannya selalu sejajar dengan sumbu fitur (axis-parallel). Untuk mengatasi hal ini, EIF diajukan sebagai alternatif yang memisahkan data melalui hyperplane dengan orientasi acak [3]. Perlu ditegaskan bahwa anomali pada penelitian ini berarti penyimpangan pola data secara statistik terhadap mayoritas trafik yang dipelajari model, bukan otomatis berarti serangan siber trafik sah yang jarang muncul dapat pula terdeteksi sebagai anomali, sebagaimana trafik menyimpang dapat tampak mirip trafik normal bila fitur yang diamati kurang membedakan keduanya.

Penelitian ini berfokus pada penerapan metode *anomaly detection* untuk mengidentifikasi penyimpangan pada trafik QUIC. Penelitian tidak berfokus pada identifikasi atau klasifikasi jenis serangan tertentu, tetapi pada kemampuan algoritma Isolation Forest (IF) dan Extended Isolation Forest (EIF) dalam membedakan trafik yang memiliki karakteristik normal dan anomali berdasarkan fitur trafik QUIC, meliputi: (1) performa pada berbagai proporsi anomali (1–10%), (2) hasil score terhadap pengaruh *Power Transfomer*, (3) sensitivitas performa terhadap berbagai *threshold,* serta (4) komputasi (memori, CPU, waktu pengujian).

II. TINJAUAN PUSTAKA

Konsep isolasi acak sebagai dasar deteksi anomali pertama kali dikemukakan oleh Liu, Ting, dan Zhou [2], tanpa bergantung pada perhitungan jarak maupun kepadatan data. Keterbatasan axis-parallel pada metode tersebut direspons oleh Hariri, Kind, dan Brunner [3] lewat EIF, yang memisahkan data melalui hyperplane berarah acak.Pada trafik TCP/UDP tidak terenkripsi, IF umumnya kompetitif: pada NSL-KDD, IF mencapai F1-score 0,993, mengungguli One-Class SVM, LOF, dan autoencoder [4]. Kombinasi IF dengan K-Means terbukti efisien untuk trafik TCP berskala industri [5].

Untuk trafik TCP/UDP yang belum terenkripsi, performa IF tergolong kompetitif pengujian pada NSL-KDD mencatat F1-

XXX-X-XXXX-XXXX-X/XX/$XX.00 ©20XX IEEE

score 0,993, melampaui One-Class SVM, LOF, maupun autoencoder [4]. Efisiensi juga ditunjukkan ketika IF digabungkan dengan K-Means pada trafik TCP skala industri [5].

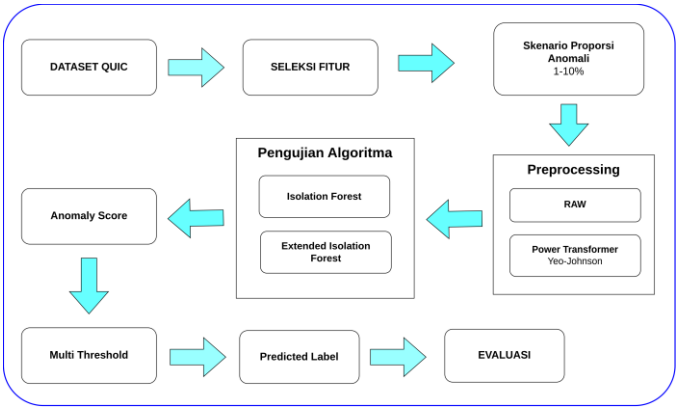
Pada ranah IoT, EIF terbukti mampu mengenali pola anomali trafik yang heterogen [7]. Namun pada log Zeek, performa terbaik EIF justru dicapai pada extension level nol setara axis-parallel IF dengan akurasi berkisar 78,3–82,3% tergantung jenis pola serangan [8], mengindikasikan bahwa superioritas EIF atas IF tidak berlaku universal. Temuan dari kajian Rotated Isolation Forest memperkuat keraguan ini dengan mengidentifikasi bentuk bias baru pada EIF berupa ghost inter-cluster [9].

Riset deteksi anomali/intrusi pada QUIC secara spesifik masih didominasi pendekatan supervised yang mengandalkan fingerprint trafik. Al-Bakhat dan Almuhammadi [10] misalnya mengidentifikasi trafik Command & Control berbahaya menggunakan enam model klasifikasi berlabel; pendekatan serupa juga diusulkan untuk mendeteksi serangan flood berbasis QUIC [11], sementara kerentanan eksfiltrasi data melalui fitur Server Preferred Address turut dievaluasi dengan pendeteksi anomali berbasis ML [12].

Dari uraian di atas terlihat celah penelitian: IF dan EIF memang sudah banyak diuji pada trafik TCP/UDP maupun IoT, tetapi penerapannya pada QUIC terenkripsi dengan skema pengujian sistematis yang mencakup variasi proporsi anomali, parameter, threshold, dan efisiensi komputasi sekaligus belum banyak dibahas. Studi QUIC yang ada pun cenderung bersifat supervised berbasis fingerprint, berbeda dari pendekatan unsupervised berbasis statistik flow yang diambil pada penelitian ini.

III. METODE PENELITIAN

Tahapan penelitian terdiri atas sembilan langkah seperti diuraikan pada subbagian A–G berikut (dapat disertai Gambar diagram alur penelitian). Dalam penelitian ini, anomali tidak secara langsung diartikan sebagai serangan, melainkan sebagai penyimpangan karakteristik data dari pola lalu lintas normal yang diamati.



A. *Dataset Publik QUIC*

Data yang dipakai berasal dari dataset publik trafik QUIC yang dipublikasikan di platform Kaggle, dikumpulkan oleh penyedia dataset menggunakan perangkat 6cure untuk merekam flow bidirectional dari server HTTP/3 pada lingkungan simulasi/terisolasi; hasil ekstraksinya tersedia dalam format .csv melalui repositori GitHub [13]. File yang digunakan, Dataset1.csv, berisi 7.330 baris flow numerik terbagi atas 3.330 baris berlabel normal dan 4.000 baris berlabel malicious, mencakup tiga skenario serangan (Scan, QUIC Connect Flood, GET Flood) — dengan total 52 kolom: 51 fitur numerik ditambah satu kolom label (0 untuk normal, 1 untuk malicious) [13].

Dalam penelitian ini, data malicious digunakan sebagai representasi data anomali untuk membentuk kondisi pengujian. Penelitian tidak berfokus pada identifikasi jenis serangan, tetapi pada kemampuan Isolation Forest (IF) dan Extended Isolation Forest (EIF) dalam mendeteksi penyimpangan pada karakteristik trafik QUIC. Label pada dataset digunakan sebagai ground truth pada tahap evaluasi, sedangkan tidak digunakan sebagai masukan algoritma.

B. *Seleksi Fitur*

Fitur yang digunakan merupakan karakteristik statistik flow QUIC yang mencakup ukuran dan proporsi paket, throughput, entropi, inter-arrival time (IAT), rasio paket, significant gaps, jumlah datagram UDP, serta statistik spin bit [13]. Sebanyak 51 fitur numerik digunakan sebagai masukan algoritma.

Fitur label tidak digunakan dalam proses pengujian karena IF merupakan metode unsupervised anomaly detection yang mendeteksi anomali berdasarkan proses isolasi data tanpa memerlukan label kelas pada proses pembentukan model [2]. Prinsip yang sama diterapkan pada EIF sebagai pengembangan Isolation Forest yang memperluas mekanisme pemisahan melalui hyperplane yang tidak harus sejajar dengan sumbu fitur [3]. Dengan demikian, label 0 dan 1 disimpan secara terpisah dan hanya digunakan sebagai ground truth pada tahap evaluasi.

C. *Proporsi Anomali*

Dataset pengujian dibentuk menggunakan data normal sebagai data dasar dan data malicious sebagai representasi anomali. Pengujian dilakukan pada sepuluh tingkat proporsi anomali, yaitu 1% hingga 10% dengan kenaikan 1%. Variasi tersebut digunakan sebagai skenario eksperimen untuk mengamati perubahan kinerja IF dan EIF ketika proporsi data anomali dalam populasi pengujian meningkat. Penggunaan beberapa tingkat proporsi memungkinkan kemampuan metode unsupervised anomaly detection diamati pada kondisi distribusi data yang berbeda [14]. Sampel anomali dipilih secara acak, dan seluruh proses pembentukan dataset skenario, pelatihan model, serta evaluasi diulang sebanyak 20 kali dengan seed acak berbeda (0 hingga 19) pada setiap kombinasi proporsi anomali dan kondisi preprocessing, untuk mengamati variabilitas hasil akibat pengambilan sampel acak dan menjaga reproducibility setiap eksperimen.

D. *Preprocessing Data*

Seluruh fitur yang digunakan dalam penelitian berupa fitur numerik sehingga tidak diperlukan tahap encoding. Data

dipersiapkan dalam dua kondisi, yaitu data tanpa transformasi (raw) dan data yang ditransformasikan menggunakan Power Transformer dengan metode Yeo-Johnson. Kondisi raw digunakan sebagai baseline, sedangkan kondisi hasil transformasi digunakan untuk mengamati pengaruh transformasi distribusi fitur terhadap kinerja IF dan EIF.

Sebelum masuk tahap pemodelan, tiap fitur diperiksa nilai skewness-nya untuk melihat seberapa menceng distribusinya. Untuk meringkas kondisi keseluruhan fitur, dihitung rata-rata dari nilai absolut skewness — nilai absolut dipilih supaya kemencengan ke arah positif dan negatif dari fitur-fitur berbeda tidak saling meniadakan ketika dirata-ratakan.

Selanjutnya, Power Transformer dengan metode Yeo-Johnson diterapkan menggunakan standardize=True. Yeo-Johnson merupakan transformasi power yang dapat digunakan pada data dengan nilai positif maupun negatif dan bertujuan menghasilkan distribusi yang lebih mendekati bentuk Gaussian [16]. Parameter standardize=True digunakan untuk melakukan standardisasi hasil transformasi.

Untuk memberikan gambaran perubahan distribusi secara visual, fitur dengan nilai absolut skewness terbesar pada kondisi raw dipilih sebagai fitur representatif. Distribusi fitur data tersebut kemudian dibandingkan sebelum dan sesudah transformasi untuk menunjukkan perubahan karakteristik distribusinya.

E. Isolation Forest dan Extended Isolation Forest

IF membangun pohon biner acak (isolation tree) dengan skor anomali berdasar rata-rata path length: semakin pendek path length, semakin mudah suatu data diisolasi dan semakin tinggi indikasi anomalinya [2]. EIF menggunakan mekanisme serupa, namun pemisahan pada tiap simpul pohon dilakukan melalui hyperplane dengan arah acak (kombinasi linear dari seluruh fitur), bukan pemisahan sejajar sumbu seperti pada IF [3]. Karena arah hyperplane bergantung pada skala relatif tiap fitur, EIF secara teoretis lebih sensitif terhadap perbedaan skala antar-fitur dibanding IF. Penelitian ini menggunakan parameter default *n_estimator(100), max_sample(256), extension_level(50)* seluruh fitur digunakan untuk menguji *hyperplane.* Dimana *n_estimator* adalah jumlah pohon isolasi dan *max_sample* adalah jumlah data yang diuji per pohon isolasi.

Keluaran kedua algoritma berupa skor anomali kontinu. Pada IF, skor didasarkan pada panjang jalur rata-rata (average path length) untuk mengisolasi suatu data, sedangkan EIF menggunakan mekanisme isolasi yang sama dengan bidang pemisah yang diperluas [2], [3]. Skor anomali dari kedua model lalu dipakai pada tahap pemilihan threshold untuk menghasilkan label prediksi.

Implementasi EIF yang dipakai di sini dibangun sendiri berbasis NumPy, bukan memakai pustaka eif yang lazim digunakan komunitas — pustaka tersebut terikat pada kompilasi C++ versi lama yang tidak kompatibel dengan versi C++ di lingkungan penelitian ini. Konsekuensinya, perbandingan waktu dan memori antara IF (scikit-learn) dan EIF (implementasi mandiri) ikut terpengaruh perbedaan level implementasi, bukan murni mencerminkan perbedaan algoritmanya. Untuk mengecek validitas implementasi mandiri tersebut, dijalankan sanity check dengan menyetel EIF pada extension_level=0, yang secara teori setara dengan pemisahan axis-parallel ala IF [3]. Pengujian ini dilakukan pada seluruh sepuluh proporsi anomali, baik pada data raw maupun data yang telah ditransformasi dengan Power Transformer (bukan hanya satu skenario tunggal), dengan menghitung korelasi Spearman antara skor EIF(extension_level=0) dan skor IF scikit-learn pada data yang sama; hasil lengkapnya disajikan pada Subbagian Hasil Pengujian IF dan EIF.

F. Strategi Pemilihan Threshold dan Label Prediksi

Konversi skor anomali kontinu menjadi label biner dilakukan dengan menguji 200 titik kuantil sebagai kandidat threshold pada tiap skenario; titik yang menghasilkan F1-score tertinggi dijadikan threshold acuan pelaporan.

G. Evaluasi

Performa model dievaluasi menggunakan *precision*, *recall*, *F1-score*, dan AUPRC. AUPRC digunakan sebagai metrik pelengkap karena lebih informatif pada kondisi *class imbalance*, terutama ketika kelas anomali merupakan kelas minoritas [15]. Selanjutnya, *threshold sweep* dilakukan untuk menentukan ambang operasional berdasarkan nilai F1-score maksimum. Evaluasi performa mengacu pada empat metrik: precision, recall, F1-score, dan AUPRC. AUPRC dipilih sebagai pelengkap karena lebih representatif ketika kelas anomali menjadi minoritas dalam kondisi class imbalance [15]. Penentuan ambang operasional dilakukan lewat threshold sweep berdasarkan F1-score maksimum, sehingga F1-score berfungsi sebagai metrik acuan utama — dipilih karena langsung mencerminkan keseimbangan precision dan recall pada threshold yang dilaporkan.

H. Lingkungan Pengujian

Semua eksperimen dijalankan di Google Colaboratory menggunakan runtime CPU tanpa akselerasi GPU/TPU. Perangkat yang dialokasikan mencakup prosesor Intel(R) Xeon(R) CPU @ 2.20GHz dengan total RAM 12 GB (kapasitas tersedia rata-rata ±10 GB saat eksperimen berjalan). Dari sisi software, digunakan Python 3.13.15, scikit-learn 1.6.1 untuk IF, serta NumPy 2.1.3 sebagai basis implementasi mandiri EIF; konsumsi memori dan CPU dipantau lewat pustaka psutil, sedangkan waktu pelatihan dan pengujian dicatat menggunakan modul time bawaan Python.

IV. HASIL DAN PEMBAHASAN

A. Hasil Pengujian IF dan EIF

Tabel I menyajikan precision, recall, dan F1-score IF dan EIF pada threshold terbaik untuk setiap proporsi anomali, tanpa Power Transformer.

TABLE I. PERBANDINGAN HASIL RATIO ANOMALI - RAW DATA

Ratio	Precision		Recall		F1-Score		AUPRC	
	IF	EIF	IF	EIF	IF	EIF	IF	EIF
1%	0.205	0.015	0.424	0.333	0.277	0.028	0.152	0.009
2%	0.325	0.032	0.417	0.373	0.366	0.059	0.207	0.206
3%	0.432	0.048	0.450	0.350	0.441	0.084	0.312	0.03

4%	0.369	0.06	0.488	0.323	0.420	0.101	0.294	0.039
5%	0.426	0.078	0.542	0.331	0.477	0.126	0.359	0.045
6%	0.479	0.087	0.470	0.320	0.474	0.137	0.323	0.051
7%	0.446	0.079	0.480	0.300	0.462	0.126	0.335	0.054
8%	0.458	0.093	0.500	0.304	0.479	0.142	0.358	0.062
9%	0.478	0.109	0.436	0.293	0.456	0.159	0.322	0.071
10%	0.455	0.090	0.504	0.239	0.478	0.131	0.365	0.073

Pada data raw, F1-score IF berada pada 0,277–0,479 (rerata 0,433), sedangkan F1-score EIF hanya 0,028–0,166 (rerata 0,113). Selisih F1 EIF terhadap IF berkisar dari −0,249 (1%) hingga −0,357 (3%), sehingga IF lebih tinggi pada seluruh sepuluh proporsi. Precision EIF berada pada 0,015–0,109 dan Recall EIF pada 0,239–0,373, artinya pada threshold terbaiknya EIF raw menandai banyak flow normal sebagai anomali. F1 IF naik dari 0,277 (1%) menjadi 0,441 (3%), kemudian berkembang pada 0,420–0,479 hingga 10%. Nilai tertinggi 0,479 (8%) hanya terpaut 0,001–0,002 dari nilai pada 5% (0,477) dan 10% (0,478), sehingga tidak ada dasar untuk menyebut satu proporsi sebagai yang terbaik. Threshold operasional untuk kondisi raw dilaporkan pada Tabel III, dan hasil pengulangan 20 kali dengan seed berbeda (Tabel V) menguatkan temuan ini: F1-score IF signifikan lebih tinggi dibandingkan EIF pada seluruh sepuluh proporsi (uji Wilcoxon signed-rank, $p<0,001$ pada setiap proporsi), sehingga keunggulan IF pada data raw bukan kebetulan sampling.

TABLE II. PERBANDINGAN HASIL RATIO ANOMALI - POWER TRANSFORMER

Ratio	Precision		Recall		F1-Score		AUPRC	
	IF	EIF	IF	EIF	IF	EIF	IF	EIF
1%	0.411	0.617	0.848	0.636	0.554	0.626	0.439	0.515
2%	0.581	0.782	0.746	0.805	0.653	0.794	0.584	0.765
3%	0.628	0.752	0.760	0.910	0.687	0.823	0.647	0.764
4%	0.729	0.583	0.669	0.917	0.698	0.713	0.683	0.701
5%	0.662	0.744	0.710	0.789	0.686	0.766	0.671	0.806
6%	0.752	0.708	0.670	0.755	0.708	0.731	0.656	0.750
7%	0.690	0.589	0.583	0.905	0.632	0.714	0.648	0.733
8%	0.675	0.610	0.687	0.830	0.681	0.703	0.658	0.720
9%	0.649	0.630	0.593	0.766	0.620	0.691	0.627	0.719
10%	0.565	0.662	0.750	0.588	0.645	0.623	0.668	0.661

Dengan PT, F1-score IF berada pada 0,554–0,708 (rerata 0,656) dan F1-score EIF pada 0,623–0,823 (rerata 0,718). EIF lebih tinggi pada sembilan dari sepuluh proporsi, dengan selisih 0,015–0,141. Selisih terbesar terdapat pada 2% (0,794 vs 0,653) dan 3% (0,823 vs 0,687), sedangkan yang terkecil pada 4% (0,015), 8% (0,022), dan 6% (0,023). Pada 10%, IF lebih tinggi 0,022 (0,645 vs 0,623). Dua catatan diperlukan. Pertama, pada 4%, 6%, 8%, dan 10% selisih F1 di bawah 0,03; karena setiap skenario dijalankan sekali, selisih sekecil itu tidak dapat dibedakan dari variasi pengambilan sampel. Kedua, AUPRC menunjukkan pola yang sama (EIF lebih tinggi pada sembilan dari sepuluh proporsi pengecualian pada 10%: 0,661 vs 0,668), sehingga arah temuan konsisten pada dua metrik yang dihitung dengan cara berbeda, yaitu F1 pada satu threshold dan AUPRC pada seluruh threshold. Threshold operasional untuk kondisi dengan PT dilaporkan pada Tabel IV. Namun, hasil pengulangan 20 kali dengan seed berbeda beserta uji Wilcoxon signed-rank (Tabel V) menunjukkan bahwa perbedaan F1-score antara IF dan EIF setelah PT TIDAK signifikan secara statistik pada seluruh sepuluh proporsi (p berkisar 0,143–0,927, semuanya >0,05). Dengan demikian, klaim "EIF unggul pada sembilan dari sepuluh proporsi" tidak dapat dipertahankan sebagai temuan yang kuat secara statistik; yang dapat disimpulkan adalah performa IF dan EIF setelah PT berada pada level yang sebanding (comparable), bukan EIF yang secara konsisten unggul.

Tabel III menyajikan threshold terbaik yang terpilih pada setiap proporsi anomali untuk kondisi raw, berdasarkan threshold dengan F1-score maksimum pada run tunggal yang sama dengan Tabel I.

TABLE III. THRESHOLD TERBAIK TERPILIH - DATA RAW

Ratio	Threshold IF	Threshold EIF
1%	0,554	0,442
2%	0,565	0,441
3%	0,549	0,432
4%	0,536	0,440
5%	0,511	0,430
6%	0,529	0,441
7%	0,511	0,433
8%	0,508	0,431
9%	0,515	0,428
10%	0,494	0,380

Tabel IV menyajikan threshold terbbaik yang terpilih pada setiap proporsi anomali untuk kondisi dengan Power Transformer, berdasarkan threshold dengan F1-score maksimum pada run tunggal yang sama dengan Tabel II.

TABLE IV. THRESHOLD TERBAIK - DATA DENGAN POWER TRANSFORMER

Ratio	Threshold IF	Threshold EIF
1%	0,591	0,576
2%	0,585	0,567
3%	0,577	0,557
4%	0,581	0,543
5%	0,569	0,555
6%	0,573	0,541
7%	0,573	0,536
8%	0,564	0,535
9%	0,556	0,538
10%	0,546	0,544

Sebagai tabel pendukung dari hasil run tunggal pada Tabel I dan Tabel II, Tabel V menyajikan hasil pengulangan eksperimen sebanyak 20 kali dengan seed acak berbeda untuk setiap kombinasi proporsi anomali dan kondisi preprocessing, beserta hasil uji Wilcoxon signed-rank berpasangan antara F1-score IF

dan EIF pada 20 pengulangan yang sama. Tabel ini menunjukkan apakah selisih F1-score yang teramati pada run tunggal (Tabel I/II) merupakan perbedaan yang konsisten secara statistik atau hanya variasi pengambilan sampel acak.

TABLE V. HASIL PENGULANGAN 20X SEED BERBEDA DAN UJI WILCOXON SIGNED-RANK (F1-SCORE)

Ratio	Kondisi	F1 IF (mean±SD)	F1 EIF (mean±SD)	p (Wilcoxon)
1%	Raw	0,241±0,107	0,020±0,011	<0,001
1%	Dengan PT	0,488±0,110	0,545±0,119	0,231
2%	Raw	0,358±0,090	0,041±0,019	<0,001
2%	Dengan PT	0,578±0,078	0,594±0,107	0,571
3%	Raw	0,412±0,088	0,067±0,018	<0,001
3%	Dengan PT	0,632±0,085	0,631±0,073	0,927
4%	Raw	0,431±0,065	0,084±0,021	<0,001
4%	Dengan PT	0,633±0,057	0,645±0,073	0,475
5%	Raw	0,452±0,046	0,107±0,020	<0,001
5%	Dengan PT	0,621±0,062	0,660±0,095	0,143
6%	Raw	0,459±0,063	0,116±0,020	<0,001
6%	Dengan PT	0,638±0,082	0,649±0,073	0,729
7%	Raw	0,463±0,047	0,133±0,019	<0,001
7%	Dengan PT	0,639±0,063	0,656±0,067	0,368
8%	Raw	0,458±0,056	0,145±0,012	<0,001
8%	Dengan PT	0,607±0,063	0,624±0,091	0,430
9%	Raw	0,482±0,057	0,153±0,013	<0,001
9%	Dengan PT	0,594±0,068	0,623±0,100	0,330
10%	Raw	0,483±0,040	0,166±0,014	<0,001
10%	Dengan PT	0,598±0,048	0,594±0,058	0,898

Sebagai validasi tambahan terhadap implementasi EIF, Tabel VI menyajikan hasil sanity check *extension_level=0* terhadap IF scikit-learn pada seluruh sepuluh proporsi anomali (1-10%), diuji pada dua kondisi data raw dan data yang telah ditransformasi dengan Power Transformer, hasilnya menunjukkan korelasi Spearman antara skor EIF(extension_level=0) dan skor IF scikit-learn konsisten tinggi pada seluruh proporsi dan kedua kondisi data: pada data raw berkisar dari rho=0,936 hingga rho=0,968 (rerata rho=0,954), dan pada data dengan PT berkisar dari rho=0,961 hingga rho=0,981 (rerata rho=0,975), dengan p-value mendekati 0 pada setiap proporsi dan kedua kondisi. Selisih F1-score terbaik antara kedua model pada pengujian ini juga kecil pada seluruh proporsi, baik raw maupun PT. Konsistensi korelasi tinggi pada seluruh rentang proporsi anomali dan pada kedua kondisi preprocessing mengindikasikan bahwa implementasi EIF mandiri berperilaku sesuai teori extension_level=0 pada kedua kondisi data, sehingga perbedaan performa EIF dibanding IF pada data raw (Tabel I) maupun kesetaraan performa keduanya setelah PT (Tabel II) merupakan karakteristik algoritma pada kondisi data tersebut, bukan kesalahan pada implementasi mandiri yang digunakan.

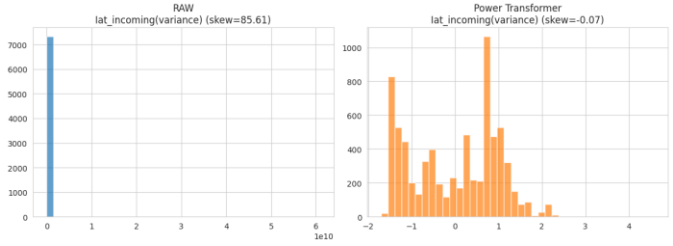
TABLE VI. SANITY CHECK EIF (EXTENSION_LEVEL=0) VS IF SCIKIT-LEARN PADA SELURUH PROPORSI ANOMALI, DATA RAW DAN DATA DENGAN POWER TRANSFORMER

Ratio	Kondisi	rho (Spearman)	p (rho)	F1 IF (sklearn)	F1 EIF (ext=0)
1%	Raw	0,952	<0,001	0,277	0,328
1%	Dengan PT	0,964	<0,001	0,554	0,452
2%	Raw	0,949	<0,001	0,366	0,353
2%	Dengan PT	0,961	<0,001	0,654	0,647
3%	Raw	0,966	<0,001	0,441	0,415
3%	Dengan PT	0,979	<0,001	0,688	0,667
4%	Raw	0,948	<0,001	0,421	0,420
4%	Dengan PT	0,976	<0,001	0,698	0,655
5%	Raw	0,954	<0,001	0,477	0,457
5%	Dengan PT	0,980	<0,001	0,686	0,723
6%	Raw	0,936	<0,001	0,475	0,499
6%	Dengan PT	0,980	<0,001	0,709	0,742
7%	Raw	0,961	<0,001	0,463	0,466
7%	Dengan PT	0,977	<0,001	0,633	0,702
8%	Raw	0,952	<0,001	0,478	0,468
8%	Dengan PT	0,981	<0,001	0,682	0,744
9%	Raw	0,968	<0,001	0,456	0,493
9%	Dengan PT	0,978	<0,001	0,620	0,635
10%	Raw	0,955	<0,001	0,479	0,511
10%	Dengan PT	0,978	<0,001	0,645	0,716

B. *Distribusi data RAW dan Power Transformer*

Untuk mengurangi kemencengan distribusi antarfitur sebelum data masuk ke tahap pemodelan, diterapkan transformasi Power Transformer dengan metode Yeo-Johnson. Efektivitasnya terlihat dari rata-rata nilai absolut skewness seluruh fitur, yang tercatat sebesar 12,9318 sebelum transformasi dan turun tajam menjadi 0,4669 sesudahnya — penurunan ini mengindikasikan bentuk distribusi fitur bergeser mendekati kesimetrisan.

Sebagai ilustrasi fitur dengan nilai absolut skewness terbesar pada kondisi raw adalah *Iat_incoming*, dengan nilai skewness sebesar 85,61. Setelah transformasi Yeo-Johnson, nilai tersebut menjadi −0,07. Perubahan distribusi *Iat_incoming* sebelum dan sesudah transformasi ditunjukkan pada Gambar 1.



Gambar 1. Ilustrasi salah satu fitur sebelum dan sesudah transformasi

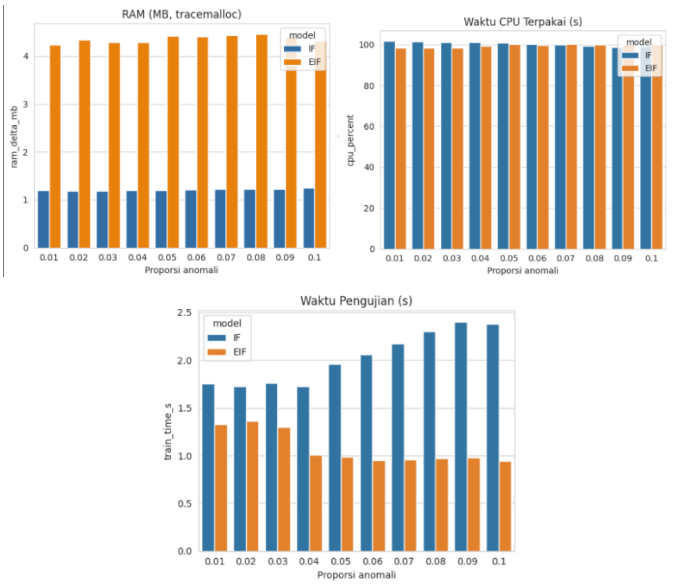
Perubahan distribusi tersebut memberikan konteks terhadap peningkatan performa EIF setelah preprocessing. EIF menggunakan hyperplane yang dibentuk dari kombinasi linear fitur, sehingga perubahan distribusi dan skala fitur dapat memengaruhi karakteristik pemisahan data. Hasil eksperimen menunjukkan bahwa setelah transformasi, F1-score EIF meningkat secara konsisten dibandingkan kondisi raw, sedangkan peningkatan IF juga terlihat pada seluruh proporsi anomali yang diuji.

C. *Pengukuran komputasi*

Rata-rata konsumsi memori, CPU, dan waktu eksekusi kedua algoritma pada keseluruhan 40 skenario pengujian dirangkum dari Tabel I dan Tabel II. Pada penggunaan memori, EIF secara konsisten membutuhkan delta RAM tiga hingga empat kali lebih besar daripada IF (rata-rata 4,36 MB berbanding 1,21 MB). Sebaliknya, pada penggunaan CPU, IF justru mencatat rata-rata yang sedikit lebih tinggi dibandingkan

EIF (100% berbanding 99,3%). Perbedaan konsumsi memori ini selaras dengan perbedaan struktur penyimpanan tiap simpul pohon pada kedua algoritma: EIF menyimpan vektor arah hyperplane berdimensi penuh (51 elemen), sedangkan IF hanya menyimpan satu indeks fitur beserta satu nilai ambang batas.

Selama waktu pengujian, EIF pada penelitian ini justru tercatat lebih cepat dibanding IF pada mayoritas skenario (rata-rata 1,01 detik berbanding 2,02 detik) berlawanan dengan ekspektasi teoretis kompleksitas komputasi EIF. Perbedaan ini diduga dipengaruhi oleh perbedaan tingkat optimasi implementasi antara IsolationForest scikit-learn dan implementasi EIF mandiri berbasis NumPy pada penelitian ini, sehingga perbandingan waktu komputasi perlu dimaknai dalam konteks tersebut.



Gambar 2. Grapik Pengukuran RAM, CPU, dan waktu pengujian

V. KESIMPULAN

Penelitian ini membandingkan performa Isolation Forest (IF) dan Extended Isolation Forest (EIF) dalam mendeteksi anomali pada trafik QUIC berdasarkan 51 fitur statistik flow, pada sepuluh proporsi anomali (1–10%), dengan dan tanpa transformasi Power Transformer metode Yeo-Johnson, divalidasi melalui 20 pengulangan seed berbeda per skenario dan uji Wilcoxon signed-rank. Pada data raw, IF secara konsisten dan signifikan mengungguli EIF pada seluruh proporsi anomali (rerata F1-score 0,424 berbanding 0,103; $p<0,001$ di setiap proporsi). Setelah diterapkan Power Transformer, rerata F1-score EIF (0,622) sedikit lebih tinggi secara numerik dibandingkan IF (0,603), namun uji Wilcoxon menunjukkan perbedaan ini tidak signifikan secara statistik pada seluruh sepuluh proporsi ($p>0,05$), sehingga klaim keunggulan EIF pada kondisi ini tidak dapat dipertahankan; yang dapat disimpulkan adalah performa IF dan EIF setelah PT berada pada level yang sebanding. Validitas implementasi EIF mandiri juga diperiksa melalui sanity check extension_level=0 terhadap IF scikit-learn pada seluruh sepuluh proporsi anomali, baik pada data raw maupun data dengan Power Transformer, yang menghasilkan korelasi Spearman tinggi dan konsisten pada kedua kondisi (data raw: rho=0,936–0,968, rerata rho=0,954; data dengan PT: rho=0,961–0,981, rerata rho=0,975), mengonfirmasi bahwa implementasi EIF konsisten dengan perilaku teoretisnya pada kedua kondisi data. Temuan ini menunjukkan bahwa performa EIF sangat bergantung pada praproses data, karena mekanisme hyperplane berarah acaknya sensitif terhadap skala antar-fitur, sedangkan IF relatif tangguh terhadap kondisi ini karena pemisahannya yang sejajar sumbu. Dari sisi komputasi, EIF membutuhkan penggunaan memori sekitar tiga hingga empat kali lebih besar dibanding IF, meskipun waktu pengujiannya justru lebih cepat pada mayoritas skenario, yang diduga lebih dipengaruhi perbedaan tingkat optimasi implementasi (scikit-learn vs implementasi mandiri berbasis NumPy) daripada kompleksitas algoritma itu sendiri. Secara umum, hasil penelitian ini menunjukkan bahwa pada data QUIC terenkripsi, IF merupakan pilihan yang lebih andal pada kondisi raw, sedangkan setelah Power Transformer, IF dan EIF memberikan performa deteksi yang sebanding, sehingga pemilihan di antara keduanya pada kondisi tersebut dapat mempertimbangkan efisiensi memori, yang secara konsisten lebih baik pada IF.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada dosen pembimbing dan rekan-rekan yang telah memberikan bimbingan, masukan, serta dukungan selama pelaksanaan penelitian dan penyusunan paper ini.

REFERENCES

[1] J. Iyengar and M. Thomson, Eds., " QUIC: A UDP-Based Multiplexed and Secure Transport," RFC 9000, IETF, 2021.

[2] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation forest," in Proc. IEEE Int. Conf. Data Mining (ICDM), 2008, pp. 413–422.

[3] S. Hariri, M. C. Kind, and R. J. Brunner, "Extended isolation forest," IEEE Trans. Knowl. Data Eng., vol. 33, no. 4, pp. 1479–1489, 2021.

[4] Y.Kumar, N. Saraswat, "Comparative Evaluation of Anomaly Detection Algorithms for Network Intrusion Detection," Springer, 2025. Available: https://link.springer.com/chapter/10.1007/978-3-032-23297-7_24

[5] M. T. R. Laskar et al, "Extending Isolation Forest for Anomaly Detection in Big Data via K-Means," arXiv:2104.13190, 2021.

[6] A. Priyanshu, S. Shastri, S. S. Medicherla, "ARLIF-IDS: Attention Augmented Real-Time Isolation Forest Intrusion Detection System," arXiv:2204.09737, 2022.

[7] S. Panja, N. Patowary, S. saha, A. Nag, "Anomaly Detection in IoT Using Extended Isolation Forest," Springer, 2023. Available: https://link.springer.com/chapter/10.1007/978-3-031-22485-0_1

[8] F. Moomtaheen, S. Bagui, S. C. Bagui, D. Mink, "Extended Isolation Forest for Intrusion Detection in Zeek Data," 2024. Available: https://www.researchgate.net/publication/382223754

[9] V. Monemizadeh, K. Kiani, "Detecting Anomalies Using Rotated Isolation Forest," arXiv:2501.17787, 2025.

[10] L. Al-Bakhat and S. Almuhammadi, "Intrusion Detection on QUIC Traffic: A Machine Learning Approach," in Proc. ICCIT, IEEE, 2022.

[11] Mr. M. V. Rajesh, et.al, "An Intelligent Machine Learning Framework for Detecting QUIC-Based Traffic Flood Attacks in Encrypted HTTP/3 Networks," 2025/2026. Available: https://doi.org/10.61137/ijsret.vol.12.issue2.193

[12] T. Gruibl, W. Niu, et.al, "QUIC-Exfil: Exploiting QUIC's Server Preferred Address Feature to Perform Data Exfiltration Attacks," arXiv:2505.05292, 2025.

[13] “QUIC Network Capture Data,” Kaggle dataset, Available: https://www.kaggle.com/datasets/adam357/quic-network-capture-data;

[14] H. Xu, G. Pang, Y. Wang, and Y.Wang, "Deep Isolation Forest for Anomaly Detection," arXiv:2206.06602, 2023.

[15] J. Davis and M. Goadrich, "The relationship between Precision-Recall and ROC curves," in *Proc. 23rd Int. Conf. Machine Learning (ICML)*, 2006, pp. 233–240, doi: 10.1145/1143844.1143874

[16] M. Yusoff, Y. Mahmud, P. A. R. Azmi, and M. T. M. Sallehud-din, "The improvement of SMOTE-ENN-XGBoost through Yeo Johnson strategy on Dissolved Gas Analysis dataset," *Energy Reports*, vol. 13, pp. 6281–6290, 2025, doi: 10.1016/j.egyr.2025.05.013.