Analisis Tingkat Risiko Keamanan Data Cloud Berdasarkan Perilaku Pengguna

Dosen Fasilitator: Prof. Kiki Ariyanti, M.Si., Ph.D.

Anggota Kelompok:

Subhan Irsyaduddien Alhaq	2306215564
Khadijah Nurul Izzah	2306153805
Muhammad Abyan Laksamana	2306210645
Irfan Hanif Yamashita	2306225943
Muhammad Daffa	2306231353
Raditya Fauzan	2306244186

Departemen Matematika
Fakultas Matematika dan Ilmu Pengetahuan Alam
Universitas Indonesia

2025

Contents

1 Pendahuluan **2**

1.1 Latar Belakang 2

1.2 Rumusan Masalah 3

1.3 Tujuan Penelitian 3

1.4 Batasan Masalah 3

2 Landasan Teori **4**

2.1 Penyimpanan Data Berbasis *Cloud* dan Risiko Keamanan 4

2.2 Log Aktivitas sebagai Sumber Informasi Keamanan 4

2.3 UEBA (*User and Entity Behavior Analytics*) 4

2.4 Deteksi Anomali untuk Keamanan Siber 5

2.5 Representasi Perilaku Harian dan *Feature Engineering* 5

2.6 Isolation Forest 6

2.7 Interpretabilitas Model dengan SHAP (*Tree SHAP*) 7

2.8 Dataset CLUE-LDS 7

3 Metodologi Penelitian **8**

3.1 Pembacaan Data dan Deskripsi Dataset 8

3.2 Eksplorasi Data dan Visualisasi 9

3.3 Transformasi Data 10

3.4 Injeksi Anomali (Pertukaran UID) 11

3.5 Pemodelan Menggunakan Isolation Forest 12

3.6 Interpretasi Model Menggunakan Tree SHAP 12

3.7 Evaluasi Kinerja Model 13

4 Hasil dan Pembahasan **14**

4.1 Hasil Prediksi Model 14

4.2 Distribusi *Anomaly Score* dan Perbandingan Label 14

4.3 Interpretasi Fitur dengan SHAP 15

4.4 Evaluasi Kinerja Model 15

5 Kesimpulan dan Saran **17**

5.1 Kesimpulan 17

5.2 Saran 17

1 Pendahuluan

1.1 Latar Belakang

Dalam era transformasi digital, layanan penyimpanan data berbasis *cloud* semakin banyak digunakan karena memungkinkan akses data secara cepat, fleksibel, dan kolaboratif dari berbagai perangkat dan lokasi. Perusahaan maupun institusi pendidikan memanfaatkan *cloud* untuk menyimpan dokumen penting, berbagi file, serta menjalankan proses kerja secara daring. Kemudahan ini membuat volume dan intensitas aktivitas pengguna di layanan *cloud* terus meningkat, sehingga keamanan data menjadi aspek yang semakin krusial.

Meskipun penyedia layanan *cloud* umumnya sudah menerapkan mekanisme keamanan teknis seperti enkripsi, kontrol akses, dan audit log, insiden keamanan tetap sering terjadi. Salah satu penyebab utamanya adalah faktor manusia, misalnya penggunaan kata sandi yang lemah, akses dari perangkat yang tidak aman, akun yang diambil alih (*account takeover*), atau aktivitas berbagi file yang tidak sesuai kebijakan. Artinya, risiko keamanan tidak hanya berasal dari kelemahan sistem, tetapi juga dari pola perilaku pengguna yang menyimpang dari kebiasaan normal.

Untuk mengatasi hal tersebut, analisis perilaku pengguna melalui log aktivitas dapat dimanfaatkan sebagai pendekatan deteksi dini ancaman. Konsep ini dikenal dalam *User and Entity Behavior Analytics* (UEBA), yaitu menganalisis kebiasaan pengguna dari waktu ke waktu untuk menemukan aktivitas yang tidak wajar, seperti lonjakan akses file, perubahan pola login, atau peningkatan aktivitas berbagi dokumen ke publik. Dengan memanfaatkan metode deteksi anomali, sistem dapat memberikan peringatan ketika suatu aktivitas berpotensi menandakan penyalahgunaan akun atau kebocoran data, bahkan ketika belum ada label serangan yang jelas.

Berdasarkan kebutuhan tersebut, penelitian ini menganalisis tingkat risiko keamanan data *cloud* berdasarkan perilaku pengguna dengan memanfaatkan dataset CLUE-LDS yang berisi log aktivitas layanan *cloud* [1]. Data log ditransformasikan menjadi ringkasan perilaku harian per pengguna [2], kemudian diterapkan model *Isolation Forest* untuk menghasilkan *anomaly score* sebagai indikator risiko [3]. Selain mendeteksi anomali, penelitian ini juga menggunakan interpretasi model melalui *Tree SHAP* untuk menjelaskan fitur perilaku yang paling berpengaruh [6], sehingga hasil analisis tidak hanya menunjukkan "anomali atau tidak", tetapi juga memberikan gambaran perilaku apa yang meningkatkan risiko keamanan.

1.2 Rumusan Masalah

Rumusan masalah penelitian ini adalah:

1. Bagaimana menganalisis risiko keamanan data *cloud* berdasarkan perilaku pengguna?
2. Apa saja jenis perilaku pengguna yang berpengaruh terhadap tingkat keamanan data pada layanan *cloud*?

1.3 Tujuan Penelitian

Tujuan penelitian ini adalah:

1. Menganalisis risiko keamanan data melalui perhitungan *anomaly score* berdasarkan perilaku pengguna.
2. Menentukan jenis perilaku pengguna yang berpengaruh terhadap tingkat keamanan data pada layanan *cloud*.

1.4 Batasan Masalah

Penelitian berfokus pada data layanan penyimpanan data *cloud*, dan hanya meninjau perilaku pengguna dalam menggunakan layanan penyimpanan data *cloud*, bukan aspek teknis infrastruktur *cloud* itu sendiri.

2 Landasan Teori

2.1 Penyimpanan Data Berbasis *Cloud* dan Risiko Keamanan

Penyimpanan data berbasis *cloud* (*cloud storage*) adalah layanan yang memungkinkan pengguna menyimpan, mengelola, dan membagikan data melalui jaringan internet. Model ini mendukung kolaborasi dan sinkronisasi lintas perangkat, sehingga proses kerja menjadi lebih cepat dan fleksibel. Namun, karakteristik utama *cloud* yang mudah diakses dari mana saja juga meningkatkan permukaan serangan, karena akun pengguna, perangkat, dan mekanisme berbagi data menjadi titik yang sering dieksploitasi.

Risiko keamanan pada *cloud storage* tidak hanya berasal dari kelemahan teknis infrastruktur, tetapi juga dari cara pengguna berinteraksi dengan sistem. Contohnya adalah penggunaan tautan berbagi yang bersifat publik, kebiasaan mengakses data pada jam yang tidak wajar, atau aktivitas akses dokumen yang tiba-tiba meningkat. Dalam konteks keamanan data, pola perilaku yang tidak sesuai dengan kebiasaan harian pengguna dapat menjadi indikasi awal terjadinya penyalahgunaan akun, kebocoran data, atau *insider threat*.

2.2 Log Aktivitas sebagai Sumber Informasi Keamanan

Layanan *cloud* umumnya menyediakan *audit log* yang merekam aktivitas pengguna, seperti percobaan login, login berhasil, akses file, pembuatan file, penghapusan, pembaruan, perubahan nama, hingga aktivitas berbagi dokumen. Log ini penting karena memberi jejak kronologis yang dapat digunakan untuk analisis forensik maupun deteksi dini ancaman.

Dari sudut pandang analitik, log aktivitas dapat dipandang sebagai rangkaian kejadian (*event stream*) yang mencerminkan perilaku pengguna. Karena aktivitas nyata pengguna cenderung memiliki pola berulang, perubahan yang signifikan pada frekuensi atau komposisi jenis aktivitas dapat digunakan untuk menandai risiko keamanan. Oleh sebab itu, pemanfaatan log sebagai data utama sangat relevan untuk memodelkan perilaku dan mendeteksi penyimpangan.

2.3 UEBA (*User and Entity Behavior Analytics*)

UEBA adalah pendekatan analisis keamanan yang berfokus pada pemodelan perilaku pengguna dan entitas (misalnya perangkat atau layanan) untuk mendeteksi aktivitas yang menyimpang dari kebiasaan normal. Inti dari UEBA adalah membangun "baseline" perilaku, lalu membandingkan aktivitas baru terhadap baseline tersebut. Pendekatan ini co-

cok untuk kasus keamanan modern karena banyak serangan (misalnya *account takeover*) terjadi dengan cara meniru aktivitas normal, tetapi biasanya tetap meninggalkan pola yang sedikit berbeda (misalnya jam akses, jenis file yang diakses, atau pola berbagi).

Pada layanan *cloud storage*, UEBA sering memanfaatkan indikator seperti lonjakan akses file, peningkatan aktivitas unduh/unggah, perubahan kebiasaan kolaborasi, atau penggunaan *public share*. Selain itu, UEBA juga sering memanfaatkan rasio aktivitas (proporsi jenis *event*) karena anomali tidak selalu muncul sebagai peningkatan jumlah aktivitas, melainkan perubahan komposisi aktivitas dalam satu periode waktu.

2.4 Deteksi Anomali untuk Keamanan Siber

Deteksi anomali adalah metode untuk menemukan observasi yang berbeda secara signifikan dari pola mayoritas data. Dalam keamanan siber, pendekatan ini penting karena label serangan seringkali tidak tersedia, tidak lengkap, atau berubah-ubah seiring munculnya teknik serangan baru. Oleh karena itu, metode *unsupervised* banyak digunakan untuk menemukan aktivitas mencurigakan tanpa perlu contoh serangan yang banyak.

Tantangan utama pada deteksi anomali untuk log keamanan adalah ketidakseimbangan data (aktivitas normal jauh lebih banyak daripada anomali) dan sifat anomali yang kontekstual (aktivitas yang aneh bagi satu pengguna bisa normal bagi pengguna lain). Akibatnya, keberhasilan deteksi tidak hanya bergantung pada model, tetapi juga pada representasi data (fitur) dan penentuan ambang keputusan (*threshold*) yang sesuai dengan kebutuhan sistem.

2.5 Representasi Perilaku Harian dan *Feature Engineering*

Agar log mudah dianalisis oleh model, data log mentah biasanya ditransformasikan menjadi representasi yang lebih ringkas. Salah satu cara yang umum digunakan adalah membentuk ringkasan perilaku harian per pengguna dengan mengelompokkan log berdasarkan pasangan (*uid*, *date*) [2]. Hasilnya, setiap baris data merepresentasikan aktivitas seorang pengguna pada satu hari, sementara kolom berisi jumlah kejadian untuk tiap jenis aktivitas (*event type*).

Selain fitur jumlah, fitur rasio juga sering digunakan, misalnya rasio akses file terhadap total aktivitas harian atau rasio aktivitas *public share*. Fitur rasio membantu menangkap perubahan pola (komposisi aktivitas), sehingga model dapat lebih peka terhadap perilaku yang "bergeser" meskipun total aktivitas tidak meningkat drastis.

2.6 Isolation Forest

Isolation Forest merupakan metode deteksi anomali *unsupervised* yang bekerja dengan prinsip bahwa data anomali cenderung lebih mudah "terisolasi" melalui pembelahan acak dibandingkan data normal [3]. Model ini membangun banyak pohon isolasi (*isolation tree*) dengan memilih fitur dan nilai pemisah secara acak. Untuk sebuah observasi, dihitung panjang jalur (*path length*) hingga observasi tersebut terisolasi; observasi yang terisolasi lebih cepat (jalur lebih pendek) cenderung dianggap anomali.

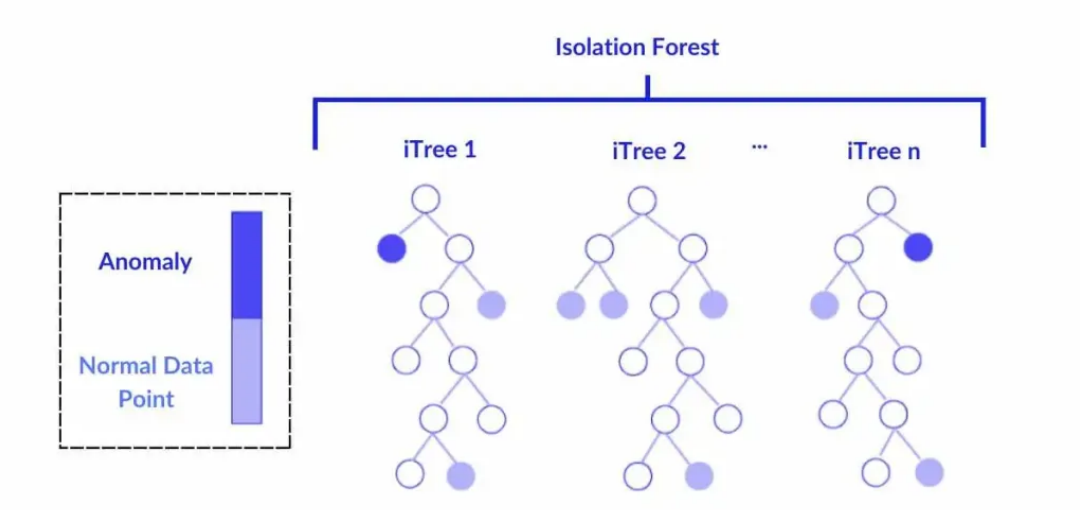

Figure 1: Isolation Forest.

Secara formal, *Isolation Tree* memiliki definisi sebagai berikut: Misalkan T adalah simpul dari *Isolation Tree*. T antara lain sebuah *external-node* (simpul daun) tanpa *child node* atau *internal-node* dengan tepat dua anak simpul (T_l, T_r). Sebuah test yang terdiri dengan atribut q dan sebuah *split value* p sedimikian sehingga test $q < p$ membagi poin-poin data menjadi T_l dan T_r.

Misal diberikan sebuah data $X = \{x_1, x_2, \ldots, x_n\}$ dari n elemen dari distribusi d-variate, untuk membangun $iTree$ ($IsolationTree$), secara recursive akan dibagi X dengan memilih secara acak atribut q dan *split value* p hingga salah satu hal berikut tercapai: (i) pohon telah mencapai batas ketinggian atau (ii) tiap simpul daun hanya memiliki satu data.

Kelebihan *Isolation Forest* adalah efisien untuk data berukuran besar dan mampu bekerja tanpa label [3]. Namun, hasilnya sensitif terhadap penentuan ambang keputusan (*threshold*) atau parameter proporsi anomali (*contamination*) [3]. Pada data yang sangat tidak seimbang, penentuan ambang yang terlalu agresif dapat meningkatkan *false positive*,

sedangkan ambang yang terlalu longgar dapat melewatkan anomali penting.

2.7 Interpretabilitas Model dengan SHAP (*Tree SHAP*)

Dalam konteks keamanan, keluaran model berupa "anomali" saja sering belum cukup karena analis perlu memahami alasan mengapa suatu aktivitas dianggap berisiko. SHAP (*SHapley Additive exPlanations*) adalah metode interpretabilitas yang menjelaskan kontribusi tiap fitur terhadap keluaran model berdasarkan konsep nilai Shapley [6]. Dengan SHAP, suatu prediksi dapat diuraikan menjadi kontribusi fitur-fitur yang mendorong skor ke arah lebih anomali atau lebih normal [6].

Untuk model berbasis pohon, termasuk *Isolation Forest*, digunakan *Tree SHAP* yang menghitung nilai SHAP secara lebih efisien pada struktur pohon [6]. Ringkasan seperti *mean absolute SHAP* dapat dipakai untuk menilai fitur mana yang paling dominan secara global, sehingga membantu menghubungkan hasil deteksi dengan perilaku nyata, misalnya peningkatan akses *public share*, pembaruan file yang tidak biasa, atau lonjakan intensitas melihat data.

2.8 Dataset CLUE-LDS

Penelitian ini menggunakan *Cloud-based User Entity Behavior Analytics Log Data Set (CLUE-LDS)* yang menyediakan log aktivitas pengguna pada layanan *cloud* untuk kebutuhan analisis UEBA [1]. Dataset ini berisi kolom penting seperti waktu kejadian (*time*), identitas pengguna anonim (*uid*), jenis aktivitas (*type*), dan label (ketika dilakukan skenario evaluasi) [1]. CLUE-LDS banyak digunakan sebagai acuan karena mencerminkan aktivitas operasional yang beragam (misalnya aktivitas login dan aktivitas file), sehingga cocok untuk studi deteksi anomali berbasis perilaku pengguna pada layanan *cloud* [1].

Selain dataset, tersedia pula referensi implementasi dan contoh pipeline pemrosesan/anomali pada CLUE-LDS yang membantu dalam replikasi eksperimen dan pemahaman alur analisis [2]. Dengan kombinasi data log, pemodelan deteksi anomali, dan interpretasi fitur, penelitian UEBA pada *cloud* dapat memberikan indikator risiko sekaligus penjelasan perilaku yang relevan untuk keamanan.

3 Metodologi Penelitian

Penelitian ini menggunakan pendekatan *anomaly detection* tanpa label (*unsupervised*) untuk menganalisis tingkat risiko keamanan pada layanan penyimpanan data *cloud* berdasarkan perilaku pengguna. Dataset yang dipakai adalah *Cloud-based User Entity Behavior Analytics Log Data Set (CLUE-LDS)* yang berisi catatan aktivitas pengguna pada layanan penyimpanan *cloud* [1]. Secara umum, penelitian ini memodelkan perilaku harian pengguna [2], menghitung *anomaly score* menggunakan *Isolation Forest* [3, 4, 5], lalu menjelaskan faktor penyebab anomali menggunakan interpretabilitas model (*Tree SHAP*) [6].

Alur penelitian ini dimulai dari pembacaan dataset [1], dilanjutkan dengan analisis eksploratif melalui visualisasi untuk memahami pola distribusi aktivitas. Setelah itu dilakukan tahap *preprocessing* dan *feature engineering* agar data log mentah berubah menjadi representasi perilaku pengguna per hari [2]. Data hasil transformasi selanjutnya diproses dengan *Isolation Forest* untuk menghasilkan *anomaly score* [3]. Terakhir, *Tree SHAP* digunakan untuk mengidentifikasi perilaku apa yang paling berkontribusi terhadap anomali [6]. Rangkuman alur dapat dilihat pada Gambar 2.

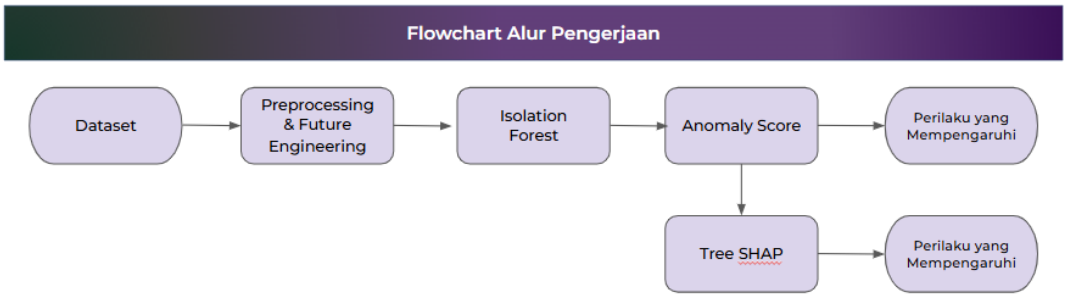

Figure 2: Flowchart alur penelitian.

3.1 Pembacaan Data dan Deskripsi Dataset

Tahap pertama dalam penelitian ini adalah mengunduh dan menggunakan dataset CLUE-LDS dari Zenodo [1]. Dataset ini memuat log aktivitas pengguna pada layanan *cloud* dan memiliki kolom penting seperti *time*, *uid*, *type*, dan *label* [1]. Mengingat volume data yang besar, pembacaan dilakukan dengan pendekatan yang lebih hemat sumber daya, seperti membaca data secara *chunk*/batch dan hanya memuat kolom yang relevan untuk analisis [2]. Strategi ini membantu mempercepat proses pemuatan data sekaligus mengurangi risiko *memory overflow* saat tahap transformasi dan pembuatan fitur. Contoh tampilan data log ditunjukkan pada Gambar 3.

	params	type	time	uid	id	role	uidType	isLocalIP	location
0	{'path': '/chinese-teal-meerkat-garagemanager/...	file_accessed	2020-12-02T10:19:40Z	ambitious-gold-bonobo-repairman	25000001	technical	name	NaN	NaN
1	{'path': '/chinese-teal-meerkat-garagemanager/...	file_accessed	2020-12-02T10:19:40Z	ambitious-gold-bonobo-repairman	25000002	technical	name	NaN	NaN
2	{'path': '/chinese-teal-meerkat-garagemanager/...	file_accessed	2020-12-02T10:19:40Z	ambitious-gold-bonobo-repairman	25000003	technical	name	NaN	NaN
3	{'path': '/chinese-teal-meerkat-garagemanager/...	file_accessed	2020-12-02T10:19:40Z	ambitious-gold-bonobo-repairman	25000004	technical	name	NaN	NaN
4	{'path': '/chinese-teal-meerkat-garagemanager/...	file_accessed	2020-12-02T10:19:40Z	ambitious-gold-bonobo-repairman	25000005	technical	name	NaN	NaN
5	{'path': '/chinese-teal-meerkat-garagemanager/...	file_accessed	2020-12-02T10:19:40Z	ambitious-gold-bonobo-repairman	25000006	technical	name	NaN	NaN

Figure 3: Tampilan data log CLUE-LDS.

3.2 Eksplorasi Data dan Visualisasi

Setelah data dibaca, dilakukan visualisasi untuk memperoleh gambaran distribusi aktivitas. Visualisasi pertama menunjukkan total aktivitas per tahun beserta proporsinya. Terlihat ketimpangan distribusi yang cukup tajam, di mana tahun 2021 menjadi pusat aktivitas utama dan menyumbang hampir separuh data [1].

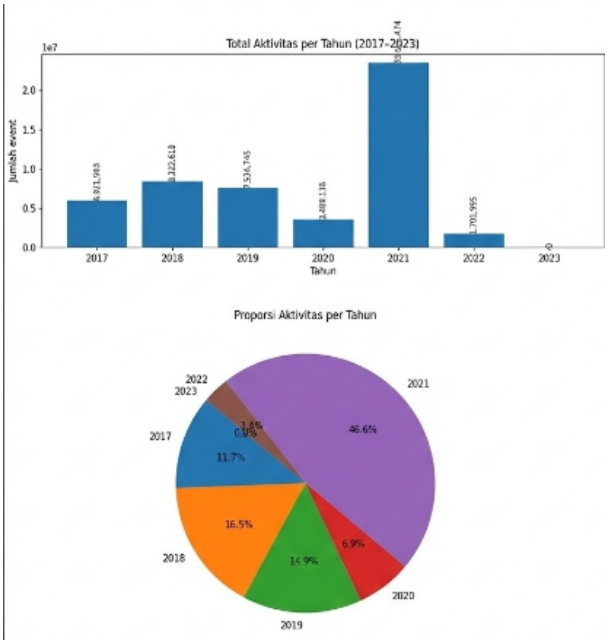

Figure 4: Distribusi total aktivitas per tahun dan proporsi aktivitas per tahun.

Visualisasi berikutnya menampilkan jumlah pengguna unik per tahun serta pola aktivitas per bulan. Meskipun jumlah pengguna unik tertinggi terjadi pada tahun 2018, lonjakan aktivitas terbesar justru terjadi pada tahun 2021. Pola ini utamanya dipengaruhi oleh lonjakan ekstrem pada bulan Maret 2021, sehingga tahun 2021 tampak sangat dominan dibandingkan tahun lain [1].

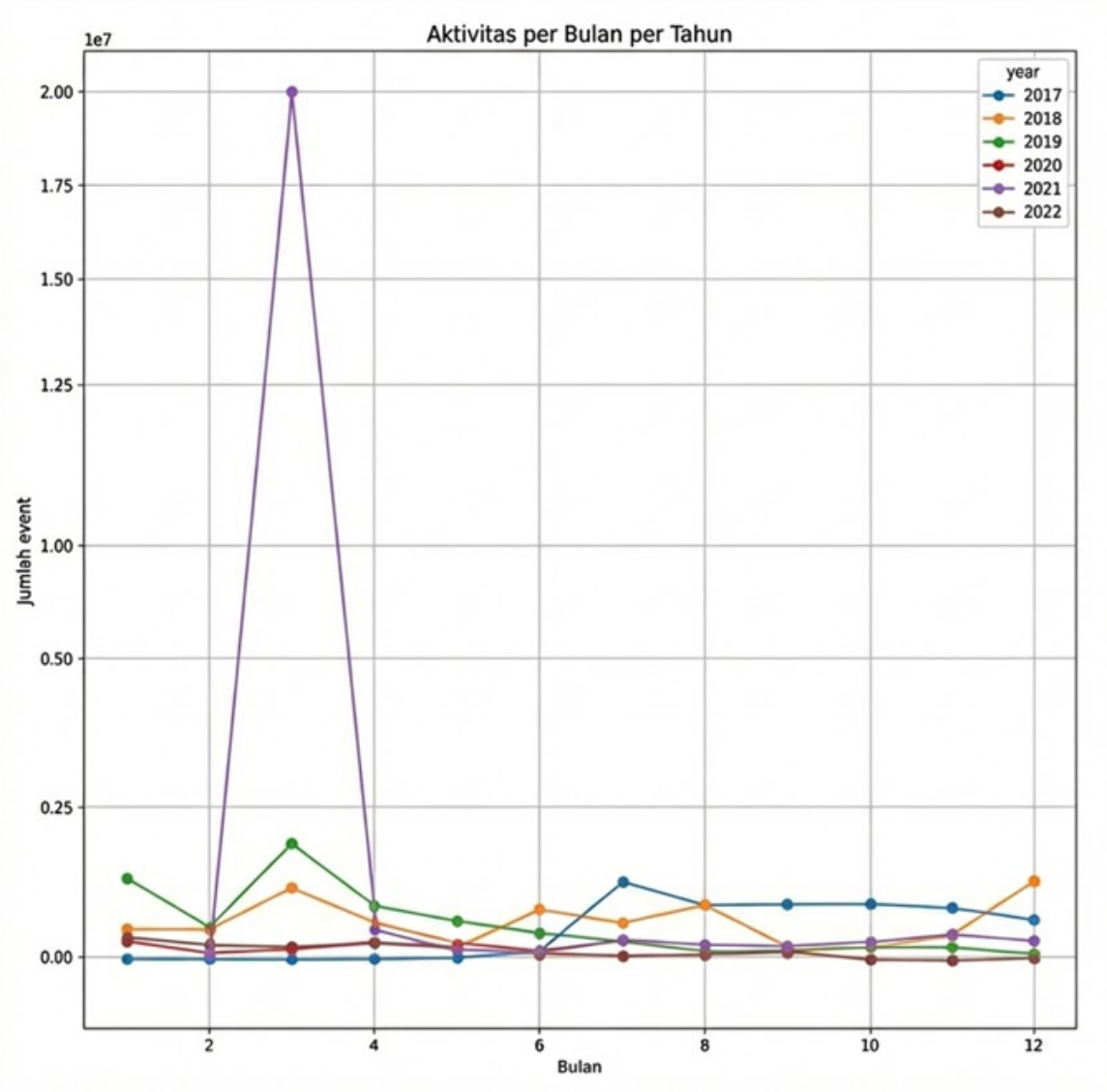

Figure 5: Jumlah pengguna unik per tahun dan aktivitas per bulan pada tiap tahun.

3.3 Transformasi Data

Data log mentah awalnya berbentuk *event* (setiap baris = satu aktivitas, berisi *time*, *uid*, dan *type*) [1]. Agar bisa dianalisis sebagai perilaku, data ditransformasikan menjadi ringkasan harian per pengguna melalui langkah berikut [2]:

1. Mengubah *time* ke format *datetime* lalu menurunkan kolom *date* dari *time*.
2. Mengelompokkan data berdasarkan pasangan (*uid*, *date*).
3. Menghitung frekuensi tiap jenis aktivitas (*type*) per hari dan membentuknya menjadi kolom fitur (mis. `file_accessed`, `file_created`, `file_deleted`, `file_renamed`, dan lain-lain).

	time	uid	type	label
0	2020-01-01 00:00:02+00:00	shared-fuchsia-cardinal-buildingadvisor	user_data_viewed	1
1	2020-01-01 00:00:02+00:00	shared-fuchsia-cardinal-buildingadvisor	user_data_viewed	1
2	2020-01-01 00:02:39+00:00	individual-chocolate-swordtail-knitter	login_attempt	1
3	2020-01-01 00:02:39+00:00	intact-gray-marlin-trademarkagent	login_successful	1
4	2020-01-01 00:02:49+00:00	individual-chocolate-swordtail-knitter	login_attempt	1

uid	date	type command_executed	deleted_from_trashbin	file_accessed	file_created	file_deleted	file_renamed
accessible-coral-ferret-repairer	2020-01-08	0	0	0	0	0	0
accessible-red-frog-technicalinstructor	2020-01-09	0	0	34	0	0	0
ancient-red-vicuna-ventriloquist	2020-01-02	0	0	24	0	0	0
	2020-01-03	0	0	35	0	0	0
	2020-01-07	0	0	32	2	2	1

Figure 6: Perubahan bentuk data dari log mentah berbasis *event* menjadi ringkasan perilaku harian per pengguna (*uid*, *date*).

3.4 Injeksi Anomali (Pertukaran UID)

Untuk kebutuhan evaluasi, anomali dibuat secara sintetis menggunakan pendekatan *swap UID* [2]. Mekanismenya dilakukan secara ringkas sebagai berikut:

1. Memilih pengguna yang memiliki aktivitas cukup banyak (berdasarkan *min_events_per_user*).
2. Memilih dua pengguna dari kandidat, misalnya u_1 dan u_2.
3. Menentukan rentang hari/kejadian dari u_1 dan u_2 yang tidak tumpang tindih (atau tumpang tindih minimal sesuai *min_overlap_days*).
4. Menukar UID pada rentang tersebut sehingga event milik u_1 diberi UID u_2 dan sebaliknya.
5. Memberi label anomali pada event hasil pertukaran, sedangkan event lainnya tetap dianggap normal.

3.5 Pemodelan Menggunakan Isolation Forest

Isolation Forest adalah metode *unsupervised* untuk mendeteksi anomali dengan prinsip bahwa data anomali cenderung lebih mudah "terisolasi" melalui pembelahan acak pada pohon dibandingkan data normal [3]. Model ini membangun banyak *isolation tree* dari sub-sampel acak. Pada setiap simpul pohon, model memilih satu fitur secara acak dan memilih nilai pemisah acak dalam rentang nilai fitur tersebut. Proses ini diulang hingga data terisolasi atau mencapai batas kedalaman [3].

Untuk sebuah observasi x, dihitung panjang jalur $h(x)$, yaitu banyaknya pembelahan dari akar sampai x terisolasi pada simpul eksternal. Observasi yang anomali biasanya memiliki $h(x)$ lebih pendek karena berada pada area yang jarang ditemui [3]. Dari kumpulan pohon, dihitung rata-rata panjang jalur $E(h(x))$ untuk menghasilkan skor anomali:

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

dengan faktor normalisasi

$$c(n) = 2H(n-1) - \frac{2(n-1)}{n}, \qquad H(i) = \sum_{k=1}^{i} \frac{1}{k}. \tag{2}$$

Secara intuitif, jika $E(h(x))$ kecil maka $s(x, n)$ cenderung lebih besar dan observasi lebih mungkin dianggap anomali [3].

Dalam implementasi, model menghasilkan *anomaly score* dan label prediksi untuk setiap pasangan (*uid*, *date*) [4, 5]. Penentuan data anomali dilakukan menggunakan ambang (*threshold*) yang dipilih. Karena data aktivitas harian umumnya didominasi normal, ambang sering ditentukan pada persentil tinggi (mis. 90–98%) dari distribusi skor agar hanya bagian ekor yang dianggap anomali. Jika diperlukan, ambang juga dapat disesuaikan agar menyeimbangkan *false positive* dan kemampuan menangkap anomali pada data evaluasi.

3.6 Interpretasi Model Menggunakan Tree SHAP

Setelah model menghasilkan skor/label anomali, langkah berikutnya adalah menjelaskan fitur apa yang paling berpengaruh pada keputusan model. Untuk itu digunakan SHAP (*SHapley Additive exPlanations*) yang menguraikan output model menjadi kontribusi tiap fitur [6]. Secara umum, output model $f(x)$ dapat ditulis sebagai:

$$f(x) = \phi_0 + \sum_{j=1}^{m} \phi_j, \tag{3}$$

dengan ϕ_0 sebagai nilai dasar (*baseline*) dan ϕ_j sebagai kontribusi fitur ke-j. Nilai ϕ_j yang besar menandakan fitur tersebut kuat mendorong output model pada observasi tersebut [6].

Karena *Isolation Forest* merupakan model berbasis pohon, perhitungan SHAP dilakukan menggunakan *Tree SHAP* yang efisien untuk struktur pohon [6]. Nilai SHAP dihitung untuk setiap fitur pada setiap observasi harian, lalu dirangkum menggunakan *mean absolute SHAP* untuk melihat fitur mana yang paling dominan secara keseluruhan. Dengan cara ini, hasil deteksi tidak hanya berupa label anomali, tetapi juga alasan utama mengapa suatu hari dianggap tidak wajar, misalnya lonjakan akses *public share*, peningkatan aktivitas pembaruan file, atau intensitas melihat data yang tidak biasa.

3.7 Evaluasi Kinerja Model

Evaluasi dilakukan dengan membandingkan label aktual dan label prediksi dalam bentuk *confusion matrix*, yang terdiri dari *true negative* (TN), *false positive* (FP), *false negative* (FN), dan *true positive* (TP). Pada penelitian ini, kelas anomali diperlakukan sebagai kelas positif. TN adalah data normal yang diprediksi normal, FP adalah data normal yang diprediksi anomali (alarm palsu), FN adalah data anomali yang diprediksi normal (anomali terlewat), dan TP adalah data anomali yang berhasil terdeteksi.

Metrik yang digunakan meliputi akurasi, presisi, *recall*, dan F1-score. Akurasi mengukur proporsi prediksi benar terhadap seluruh data, sedangkan presisi dan *recall* khusus menilai kualitas prediksi untuk kelas positif (anomali). F1-score digunakan sebagai ringkasan karena menyeimbangkan presisi dan *recall*. Rumus metrik dituliskan sebagai berikut:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}, \tag{4}$$

$$\text{Precision} = \frac{TP}{TP + FP}, \qquad \text{Recall} = \frac{TP}{TP + FN}, \tag{5}$$

$$\text{F1-score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}. \tag{6}$$

4 Hasil dan Pembahasan

4.1 Hasil Prediksi Model

Model *Isolation Forest* menghasilkan *anomaly_score* untuk setiap pasangan (*uid*, *date*) serta label prediksi [3]. Pada keluaran mentah model, nilai **1** menandakan *normal* dan **-1** menandakan *anomali*. Agar konsisten dengan label data (0 = normal, 1 = anomali), pada tahap evaluasi label model dipetakan menjadi 0 dan 1. Contoh hasil prediksi ditunjukkan pada Tabel 1.

Table 1: Contoh hasil prediksi model pada beberapa pasangan (*uid*, *tanggal*).

uid	**date**	**anomaly_score**	**label data**	**label model**
accessible-coral-ferret-repairer	2020-01-08	-0.216862	1	1
accessible-red-frog-technicalinstructor	2020-01-09	-0.138331	1	-1
ancient-red-vicuna-ventriloquist	2020-01-02	-0.193722	1	1
ancient-red-vicuna-ventriloquist	2020-01-03	-0.152302	1	-1
ancient-red-vicuna-ventriloquist	2020-01-07	-0.111974	1	-1

4.2 Distribusi *Anomaly Score* dan Perbandingan Label

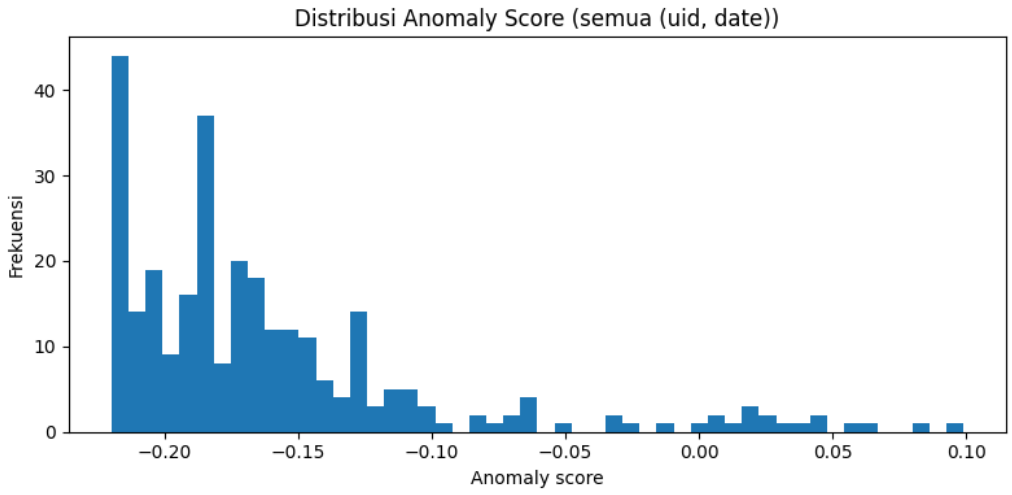

Figure 7: Distribusi *anomaly score* untuk seluruh pasangan (*uid*, *date*).

Berdasarkan Gambar 7, sebaran *anomaly score* terlihat menumpuk di bagian awal (nilai negatif) dan semakin sedikit ketika bergerak ke nilai yang lebih tinggi, sehingga hanya sebagian kecil data yang memiliki skor ekstrem [3]. Pada data awal terdapat 243 data berlabel normal (0) dan 49 data berlabel anomali (1). Setelah dimasukkan ke model, hasil prediksi menjadi 165 data normal dan 127 data anomali. Ini menunjukkan model

menandai anomali lebih banyak dibanding label data awal, sehingga kemungkinan masih ada cukup banyak data normal yang ikut terbaca sebagai anomali, terutama jika ambang keputusan atau parameter model belum disesuaikan.

4.3 Interpretasi Fitur dengan SHAP

Untuk melihat aktivitas apa yang paling membuat model menilai suatu hari sebagai tidak wajar, digunakan SHAP [6]. Secara sederhana, semakin besar nilai *mean_abs_shap*, semakin besar pula pengaruh fitur tersebut terhadap keputusan model. Hasil pada Tabel 2 menunjukkan bahwa faktor yang paling dominan adalah akses *public share* (`ratio_public_share_accessed`), pembaruan file (`file_updated`), dan intensitas melihat data (`user_data_viewed`). Pola ini masuk akal dari sisi keamanan, karena lonjakan akses *public share* atau aktivitas melihat/memperbarui file yang tidak biasa sering menjadi sinyal awal adanya penyalahgunaan akun atau upaya pengambilan data.

Table 2: Ringkasan fitur teratas berdasarkan SHAP.

Fitur	**MeanAbsSHAP**	**MeanSHAP**	**StdAbsSHAP**
`ratio_public_share_accessed`	0.178665	0.024135	0.216297
`file_updated`	0.161853	-0.027343	0.138183
`user_data_viewed`	0.156994	-0.030463	0.238311
`ratio_file_accessed`	0.150910	0.002423	0.123388
`ratio_file_written`	0.147411	0.000923	0.148594
`ratio_user_data_viewed`	0.146058	-0.017767	0.226938
`file_written`	0.131946	-0.019843	0.106476
`file_accessed`	0.120137	-0.002526	0.142654
`ratio_login_successful`	0.119244	-0.006879	0.168323
`ratio_file_created`	0.118995	-0.003781	0.181091
`ratio_file_updated`	0.113101	-0.016495	0.139484

4.4 Evaluasi Kinerja Model

Evaluasi dilakukan pada 292 data harian (243 normal dan 49 anomali). Dari confusion matrix (Gambar 8), model berhasil mengklasifikasikan 147 data normal dengan benar, tetapi 96 data normal justru terbaca sebagai anomali (alarm palsu). Untuk kelas anomali, model berhasil menangkap 31 data anomali, namun masih melewatkan 18 data anomali yang diprediksi normal. Secara umum akurasi model adalah 0.6096. Walaupun kemampuan menangkap anomali cukup terlihat dari recall anomali 0.6327, precision anomali rendah (0.2441) dan F1-score anomali hanya 0.3523, yang menunjukkan model masih terlalu sering menandai data normal sebagai anomali sehingga hasilnya belum cukup stabil untuk dijadikan alarm keamanan tanpa penyesuaian lebih lanjut.

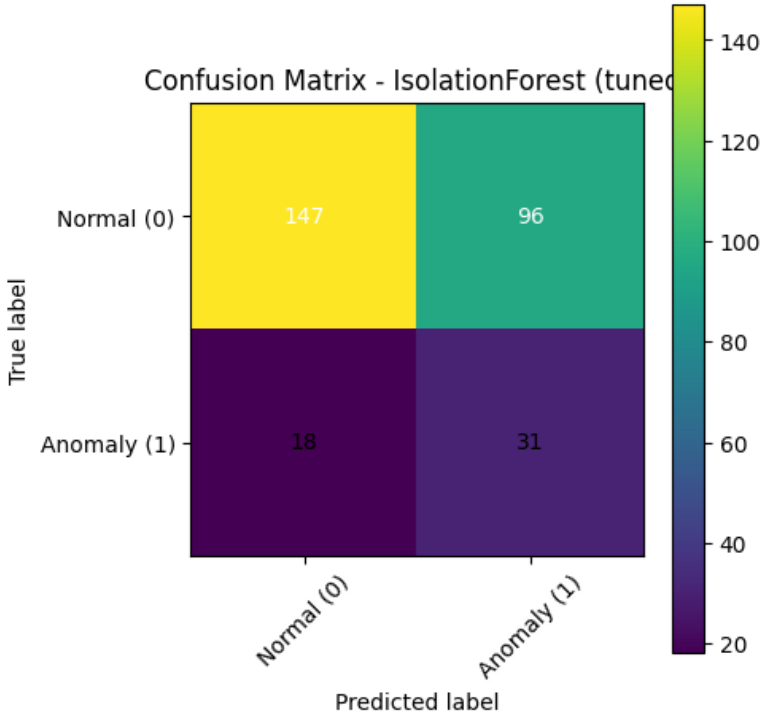

Figure 8: Confusion matrix hasil prediksi *Isolation Forest*.

Table 3: Ringkasan metrik evaluasi model.

Kelas	**Precision**	**Recall**	**F1-score**	**Jumlah data**
Normal (0)	0.8909	0.6049	0.7206	243
Anomali (1)	0.2441	0.6327	0.3523	49
Akurasi		0.6096 (292 data)		

5 Kesimpulan dan Saran

5.1 Kesimpulan

Berdasarkan penelitian yang dilakukan, analisis risiko keamanan data *cloud* dapat dilakukan dengan memodelkan perilaku pengguna dari log aktivitas menjadi ringkasan harian per pengguna [2], kemudian menerapkan metode deteksi anomali *unsupervised* menggunakan *Isolation Forest* [3]. Model menghasilkan *anomaly score* sebagai indikator tingkat kewajaran aktivitas, sehingga aktivitas harian dapat dipetakan ke kategori risiko (misalnya normal atau mencurigakan) tanpa membutuhkan pelabelan anomali sejak awal. Dengan demikian, rumusan masalah mengenai cara menganalisis risiko keamanan berdasarkan perilaku pengguna dapat dijawab melalui alur transformasi log $\rightarrow$ pemodelan $\rightarrow$ skor/label anomali.

Selain menghasilkan deteksi, penelitian ini juga mengidentifikasi jenis perilaku yang paling berpengaruh terhadap tingkat keamanan layanan *cloud* melalui interpretasi *Tree SHAP* [6]. Hasilnya menunjukkan bahwa aktivitas terkait akses *public share*, pembaruan file, serta intensitas melihat data (*user_data_viewed*) beserta proporsinya menjadi faktor dominan yang mendorong model menilai suatu hari sebagai tidak wajar. Namun, evaluasi kinerja memperlihatkan bahwa performa deteksi anomali masih belum optimal (F1-score anomali sekitar 0.3523) karena masih banyak data normal yang terbaca sebagai anomali, sehingga diperlukan penyesuaian ambang/parameter dan pengayaan fitur perilaku agar sistem lebih stabil untuk digunakan sebagai peringatan risiko keamanan.

5.2 Saran

Beberapa saran perbaikan:

1. Alih-alih menggunakan threshold statis berbasis persentil, disarankan untuk memodelkan distribusi anomaly score menggunakan pendekatan statistik, seperti fitting distribusi (misalnya Gaussian atau Gamma) atau menggunakan Gaussian Mixture Model (GMM). Metode ini memungkinkan penentuan titik potong yang lebih presisi dengan meminimalkan irisan antara distribusi data normal dan anomali, sehingga diharapkan dapat menurunkan tingkat false positive secara signifikan dan meningkatkan skor F1.
2. Penanganan ketidakseimbangan data (mis. *reweighting* atau *sampling*).
3. Evaluasi silang per periode waktu untuk mengurangi bias dominasi tahun tertentu.

References

[1] Landauer, M., Skopik, F., Höld, G., & Wurzenberger, M. (2022). *Cloud-based user entity behavior analytics log data set (CLUE-LDS)* [Data set]. Zenodo. https://doi.org/10.5281/zenodo.7119953

[2] Anomaly detection with the Cloud-based UEBA Log Data Set (clue-lds) [Computer software]. GitHub. https://github.com/ait-aecid/clue-lds

[3] Liu, F. T., Ting, K. M., & Zhou, Z.-H. (2008). *Isolation Forest*. In Proceedings of the 2008 Eighth IEEE International Conference on Data Mining (ICDM), pp. 413–422. IEEE.

[4] Tripathi, S. (2024, September 26). *Isolation Forest guide: Explanation and Python implementation*. DataCamp. https://www.datacamp.com/tutorial/isolation-forest

[5] K, D., Skelton, J., & Mukherjee, S. (2025, August 4). *Anomaly detection in Python with Isolation Forest*. DigitalOcean Community Tutorials. https://www.digitalocean.com/community/tutorials/anomaly-detection-isolation-forest

[6] Lundberg, S. M., & Lee, S.-I. (2017). *A unified approach to interpreting model predictions*. In Advances in Neural Information Processing Systems (Vol. 30).