Aplikasi Graf dalam menganalisis Jejak Digital untuk Mendeteksi Anomali pada Keamanan Informasi

Aland Mulia Pratama - 13522124[1]
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
[1]*13522124@mahasiswa.itb.ac.id*

***Abstrak*—Dalam era digital, jejak digital dan keamanan informasi menjadi aspek yang kritis. Makalah ini bertujuan untuk merumuskan metode untuk meningkatkan keamanan informasi serta jejak digital. Penerapan teori graf digunakan untuk membahas struktur dan pola koneksi jejak digital dalam representasi graf. Basis data jejak digital bermodelkan graf mempermudah divisi keamanan informasi dalam mendeteksi anomali pada jejak digital. Algoritma clustering dibutuhkan sebagai penunjang dalam mendeteksi anomali pada visualisasi jejak digital menggunakan graf. Kebutuhan terhadap algoritma disebabkan oleh kemampuan visual manusia yang terbatas dalam mendeteksi anomali pada visualisasi graf.**

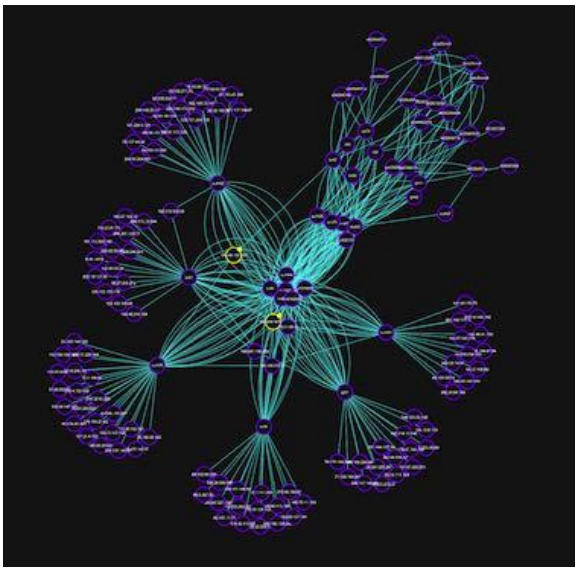
***Kata Kunci*— Graf, Keamanan Informasi, Jejak Digital, Algoritma Clustering**

I. PENDAHULUAN

Dalam era digital yang terus berkembang, jejak digital dan keamanan informasi menjadi salah satu aspek yang membutuhkan perhatian khusus. Penggunaan teknologi oleh Masyarakat yang terus meningkat menciptakan jejak digital yang luas mencakup data dan aktivitas secara daring. Seiring dengan pertumbuhan jejak digital, muncul ancaman yang berbahaya yang biasa dikenal *cyber threats* seperti Serangan siber, pencurian identitas, dan kebocoran data. Keamanan informasi yang lemah dapat menyebabkan terjadinya *cyber threats*.

Cyber threats dapat memberikan dampak yang serius terhadap beberapa entitas mulai dari individu, organisasi atau perusahaan, bahkan nasional. Salah satu contoh *cyber threats* terhadap nasional adalah kebocoran dokumen-dokumen digital yang merupakan rahasia negara. Oleh karena itu, dibutuhkan suatu penanganan khusus seperti contohnya peningkatan perlindungan siber.

Perlindungan siber merupakan suatu upaya untuk menjaga sistem, jaringan, dan perangkat lunak dari potensi ancaman digital. Biasanya, perlindungan siber terdiri dari sejumlah lapisan keamanan yang tersebar di berbagai tingkat, termasuk komputer, jaringan, perangkat lunak, dan data yang ingin dijaga oleh pengguna. Keberadaan perlindungan siber menciptakan lingkungan internet yang lebih aman bagi semua pengguna.



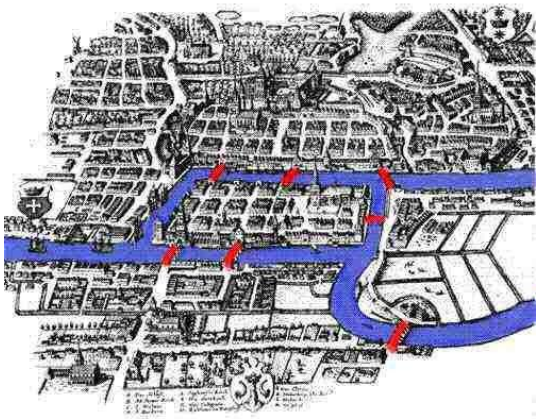
Gambar 1. contoh visualisasi data untuk keamanan siber menggunakan graf
(Sumber : https://cambridge-intelligence.com/use-cases/cybersecurity/)

Dengan menggunakan konsep graf, jaringan sosial online dan jejak digital dapat divisualisasikan ke dalam representasi graf dan memungkinkan analisis mendalam dibandingkan dengan data mentah. Jejak digital direpresentasikan dengan cara memodelkan entitas sebagai simpul dan koneksi antara entitas sebagai tepi dalam graf. Representasi jejak digital ke dalam model graf dapat memudahkan dalam mengidentifikasi risiko keamanan informasi dan juga mendeteksi perilaku anomali dalam jejak digital.

II. DASAR TEORI

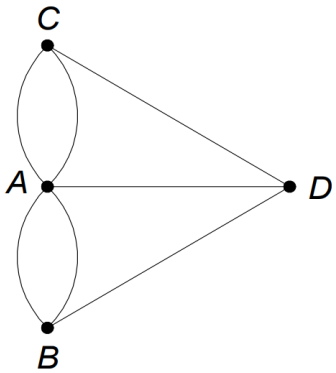
A. Graf

Graf digunakan untuk merepresentasikan objek-objek diskrit dan hubungan antara objek-objek tersebut. Graf mulai dikenal pada tahun 1736 saat Leonhard Euler memecahkan masalah mengenai jembatan Königsberg. Masalah jembatan Königsberg adalah apakah bisa orang melalui jembatan tepat sekali dan kembali ke tempat yang sama.



Gambar 2. Persoalan Jembatan Königsberg
(Sumber : maa.org)

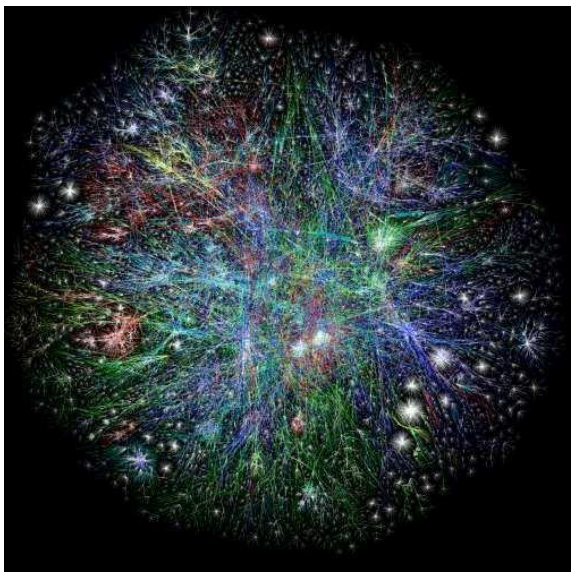
Dalam permasalahan ini Leonhard Euler (15 April 1707 – 18 September 1783) mencoba untuk memvisualisakan permasalahan tersebut. Permasalahan tersebut direpresentasikan ke dalam graf dengan simpul (*vertex*) menyatakan daratan dan sisi (*edge*) menyatakan jembatan.



Gambar 3. Visualisasi Jembatan Königsberg dalam representasi Graf

Dalam representasi ini dapat dilihat bahwa simpul (vertex) A, B, C, dan D merepresentasikan daratan dan setiap simpul yang menghubungkan tiap daratan/simpul merepresentasikan sebagai tujuh Jembatan Königsberg. Dalam pemecahan masalah ini, Leonhard Euler mengemukakan dua teori yaitu lintasan Euler (lintasan yang melalui masing-masing sisi di dalam graf tepat satu kali) dan juga sirkuit Euler (Sirkuit yang melewati masing-masing sisi tepat satu kali). Graf dengan sirkuit Euler memiliki syarat yaitu tiap simpul berderajat genap sedangkan untuk graf dengan lintasan Euler memiliki simpul berderajat ganjil dua atau tidak sama sekali. Dengan teorema ini, Leonhard Euler berhasil membuktikan bahwa orang tidak dapat melalui jembatan tepat sekali dan Kembali ke tempat yang sama tetapi setiap orang dapat melalui tiap jembatan tepat sekali.

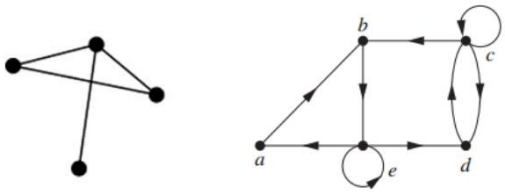
Dengan mengambil inspirasi dari konsep lintasan Euler, kita dapat membayangkan suatu situasi di mana setiap koneksi atau jalur data di internet dilewati tepat satu kali dalam suatu periode. Pemikiran ini dapat memberikan beberapa wawasan tentang manajemen jaringan dan pengaliran data dalam skala besar. Seiring dengan pertumbuhan dan evolusi teknologi, jaringan internet modern melibatkan ribuan, bahkan jutaan simpul, dan jalur koneksi yang sangat kompleks.



Gambar 4. Visualiasi Jalur data Internet dengan Graf
(Sumber : https://www.researchgate.net)

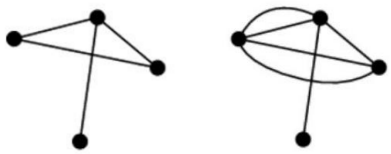
Graf merupakan salah satu bentuk struktur data yang terdiri atas simpul dan sisi. Struktur data graf memiliki notasi G(V,E). Pada umumnya, graf digunakan untuk merepresentasikan objek diskrit serta hubungan antara objek tersebut. Berdasarkan sejarah graf, graf memiliki tujuan untuk memvisualisasikan objek yang abstrak.

Graf dapat dibedakan menjadi dua jenis berdasarkan arahnya yaitu, graf berarah (*directed graph*) dan juga graf tidak berarah (*undirected graph*). Graf berarah merupakan graf yang sisinya memiliki arah sedangkan graf tidak berarah adalah graf yang sisinya tidak memiliki arah.



Gambar 5. Graf Berarah dan Graf Tidak Berarah

Graf juga dapat dibedakan ke dalam dua jenis berdasarkan ada atau tidaknya gelang, yaitu graf sederhana (*simple graph*) dan graf tidak sederhana (*unsimple graph*). Graf sederhana adalah graf yang tidak mengandung gelang sedangkan graf tidak sederhana adalah graf yang mengandung gelang.



Gambar 6. Graf Sederhana dan Graf Tidak Sederhana

Setiap simpul dari graf memiliki sesuatu yang disebut sebagai derajat. Derajat suatu simpul merupakan jumlah sisi yang terhubung dengan simpul tersebut. Derajat suatu simpul memiliki notasi d(v) dengan v sebagai simpul. Simpul dapat dikatakan simpul terpencil apabila simpul tersebut memiliki derajat nol atau tidak memiliki sisi yang terhubung dengan simpul tersebut. Berdasarkan Lemma Jabat Tangan, jumlah

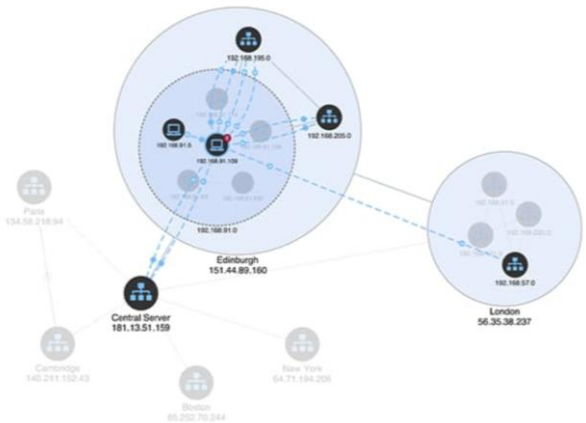
derajat semua simpul pada suatu graf adalah genap, yaitu dua kali jumlah sisi pada graf tersebut. Akibat dari lemma Jabat Tangan maka dapat disimpulkan bahwa untuk sembarang graf G banyaknya simpul berderajat ganjil selalu genap. Berdasarkan kesimpulkan tersebut, tidak mungkin sebuah graf memiliki simpul berderajat ganjil sejumlah ganjil.

Graf memiliki beberapa terminologi antara lain adalah ketetanggaan, bersisian, lintasan, sirkuit dan upagraf. Dua buah simpul dapat dikatakan terhubung atau bertetanggaan apabila terdapat sisi yang menghubungkan kedua simpul tersebuat. Selanjutnya terdapat terminologi bersisian yang berarti sebuah sisi $e = (v_j, v_k)$ dapat dikatakan bahwa sisi e bersisian dengan v_j simpul atau v_k. Berikutnya ada terminology dalam graf yaitu lintasan. Lintasan adalah barisan simpul dan sisi untuk menghubungkan suatu simpul awal dan simpul akhir. Terminology yang memiliki hubungan erat dengan lintasan adalah sirkuit yang merupakan lintasan yang berawal dan berakhir pada simpul yang sama. Terminologi terakhir yang akan dibahas merupakan upagraf. Upagraf merupakan bagian dari suatu graf dan sebuah upagraf G1 dapat dikatakan bagian dari graf G apabila simpul dan sisi dari G1 merupakan subset dari himpunan simpul dan sisi graf G.

B. Jejak Digital

Jejak digital merujuk pada informasi maupun data yang dihasilkan oleh aktivitas online dan interaksi digital seseorang. Konsep jejak digital memiliki hubungan yang erat dengan teori graf. Graf digunakan dalam analisis dan visualisasi hubungan antara entitas digital seperti pengguna internet, situs web, penyedia layanan internet, aplikasi, atau elemen-elemen lainnya.

Graf dapat digunakan untuk merepresentasikan jejak digital dengan simpul mewakili pengguna internet atau entitas digital lainnya. Tepi atau sisi dalam graf dapat merepresentasikan interaksi atau koneksi antara entitas digital. Salah satu bentuk graf yang relevan dalam jejak digital adalah graf sosial. Dalam graf sosial, pengguna internet diwakili sebagai simpul dan koneksi antara mereka diwakili sebagai tepi. Hubungan antara pengikut dan mengikuti dalam aplikasi media sosial Instagram dapat direpresentasikan dengan graf yang terhubung lemah sedangkan hubungan pertemanan dalam media sosial Facebook dapat direpresentasikan dengan graf yang terhubung kuat.



Gambar 7. Visualisasi Jejak Digital Sederhana Dengan Graf
(Sumber : https://cambridge-intelligence.com/graph-visualization-software/)

Setiap tindakan yang dilakukan oleh pengguna internet menciptakan jejak digital. Jejak digital dapat tercipta melalui aktivitas pencarian, interaksi media sosial, atau pembelian online. Hubungan antara penyebab jejak digital dan keterkaitan temporal merupakan jejak pengguna yang dapat direpresentasikan ke dalam graf dengan simpul yaitu penyebab jejak digital dan sisi atau tepi yaitu keterkaitan temporal.

Dalam konteks keamanan siber, analisis graf jejak digital dapat membantu dalam mendeteksi potensi ancaman atau serangan siber. Selain itu, perlindungan privasi menjadi perhatian penting, dan pengelolaan jejak digital perlu memperhatikan bagaimana data jejak tersebut diambil, disimpan, dan digunakan.

C. Keamanan Informasi

Graf dapat digunakan untuk menggambarkan hubungan antara berbagai elemen dalam sistem keamanan informasi. Hal ini dapat membantu analis keamanan untuk memahami sistem secara lebih baik dan lebih cepat, serta untuk mengidentifikasi potensi kerentanan, serangan, atau anomali. Dalam ranah keamanan informasi, graf dapat dimanfatkan untuk berbagai tujuan antara lain:

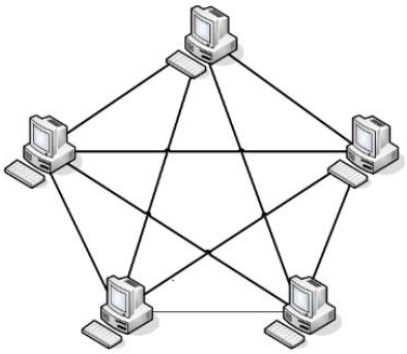
1. Visualisasi
   Graf dapat digunakan untuk memvisualisasikan hubungan antara berbagai elemen dalam sistem keamanan informasi. Visualisasi dapat membantu analisis keamanan serta mendeteksi anomali pada sistem keamanan.

2. Analisis
   Graf dapat memudahkan analisa pada sistem keamanan informasi dengan memanfaatkan visualisasi sesuai dengan tujuan graf dalam keamanan informasi. Analisis ini dapat digunakan untuk mengidentifikasi potensi kerentanan, serangan, anomali, atau ancaman siber. Ancaman siber dapat digambarkan sebagai perubahan dalam pola hubungan antara berbagai elemen dalam sistem keamanan informasi.

3. Analisis Log
   Log dapat dianalisis menggunakan graf untuk mengidentifikasi potensi serangan. Misalnya, log dari server web dapat dianalisis menggunakan graf untuk mengidentifikasi serangan Distribute Denial of Service (DDoS). Data log dari server web dapat diwakili sebagai graf di mana setiap node mewakili elemen dalam sistem seperti IP address, URL, atau jenis permintaan. Edge mungkin mewakili koneksi atau interaksi antar elemen. Korelasi antara aktivitas yang mencurigakan dan waktu kejadian dapat menjadi indikator penting dalam deteksi serangan.

4. Pemodelan Jaringan Komputer
   Jaringan pada computer dapat dimodelkan sebagai graf. Pemodelan graf pada jaringan komputer dapat direalisasikan dengan perangkat sebagai simpul / *vertex* dan koneksi sebagai sisi / *edge*. Visualisasi ini dapat membantu menganalisis serangan *man-in-the-middle*.

Dapat disimpulkan bahwa jaringan komputer yang divisualisasikan menggunakan graf merupakan graf lengkap seperti pada gambar 8.



Gambar 8. Visualisasi Jaringan Komputer menggunakan Graf

Kemampuan graf untuk memvisualisasikan hubungan yang kompleks akan sangat berguna untuk memvisualisasikan hubungan antara perangkat, pengguna, dan data dalam suatu sistem. Graf juga dapat digunakan untuk analisis yang lebih akurat. Perubahan pola dalam visualisasi graf akan lebih mudah terdeteksi dan mengidentifikasi ancaman siber. Graf dapat digunakan untuk mendeteksi serangan yang lebih canggih, seperti serangan yang memanfaatkan hubungan antara berbagai elemen dalam suatu sistem.

D. Algoritma Clustering

Clustering adalah suatu proses di mana data dikelompokkan ke dalam kelompok-kelompok yang serupa berdasarkan kesamaan karakteristik tertentu. Algoritma ini dikenalkan oleh James MacQueuen pada tahun 1967 saat ia mengembangkan algoritma ini untuk mengelompokkan data ke dalam kelompok-kelompok berdasarkan pusat kelompok. Algoritma clustering seringkali digunakan dalam ranah *machine learning* sebagai fitur tambahan pengelompokkan data.

Algoritma ini bertujuan untuk mencari kesamaan antar objek yang dibagi menjadi dua yaitu kesamaan internal dan kesamaan eksternal. Kesamaan internal dalam konteks clustering mengukur sejauh mana objek-objek di dalam satu kelompok (cluster) serupa atau homogen sedangkan kesamaan eksternal berkaitan dengan seberapa baik objek-objek dari kelompok yang berbeda benar-benar berbeda satu sama lain.

Terdapat beberapa algoritma clustering seperti K-means, *Hierarchical Clustering*, dan *Density-Based Spatial Clustering of Applications with Noise* (DBSCAN). Pada makalah ini, algoritma clustering yang akan dibahas lebih spesifik adalah K-means dan DBSCAN. Kedua algoritma tersebut akan digunakan untuk mendeteksi anomali pada data jejak digital yang diberikan.

Algoritma K-means sangat diperlukan untuk melakukan pengelompokkan data jejak digital yang diberikan. Proses pertama dalam algoritma ini adalah penentuan jumlah kelompok data. Setelah penentuan jumlah kelompok, akan dilakukan inisialisasi pusat kelompok secara acak dan untuk setiap titik data, tentukan kelompok yang memiliki pusat terdekat. Hitung ulang pusat kelompok berdasarkan rata-rata anggota kelompok. Lakukan pengulangan hingga pembaruan pada pusat kelompok tidak memberikan perubahan yang berarti dalam penempatan titik data ke dalam kelompok.

Setelah dilakukan pengelompokan data menggunakan algoritma K-means, kita akan mendeteksi anomali menggunakan algoritma DBSCAN berdasarkan kelompok data yang terbentuk. Algoritma ini diinisialisasi dengan cara memilih titik data yang belum dikunjungi. Proses berikutnya adalah menentukan apakah titik tersebut merupakan bagian dari suatu kelompok berdasarkan kepadatan lokal. Jika sebuah titik adalah *core point*, maka titik yang bertetanggaan akan dianggap sebagai bagian dari kelompok yang sama. Lakukan proses algoritma DBSCAN ini secara berulang hingga seluruh titik data telah dikunjungi. Titik data yang tidak terkait dengan kelompok manapun akan dianggap sebagai *outliers* atau anomali.

III. Aplikasi Graf dalam Visualisasi Jejak Digital

Divisi keamanan siber dalam suatu perusahaan atau organisasi pastinya memiliki banyak data seperti catatan pengguna, Alamat IP, perangkat, jaringan, dan juga server. Data ini biasanya didapat melalui alat-alat yang digunakan untuk merekam setiap jejak digital dalam entitas tersebut. Data yang didapat biasanya tidak terstruktur dan memerlukan waktu yang lama dalam menganalisa data tersebut. Namun, tantangan tersebut dapat diatasi dengan memvisualisasikan volume data yang besar dan tidak terstruktur menggunakan program. Kita dapat mengubah data yang berbasis *text* atau tulisan dengan basis data menggunakan representasi graf. Representasi graf dapat memberikan kemudahan dalam mengolah data meskipun volumenya terus berkembang.

Pemodelan jejak digital dengan representasi graf akan mengacu pada jenis data seperti pengguna, perangkat, Alamat IP, Entitas yang diakses, serta server yang digunakan. Lalu, hubungan antardata tersebut direpresentasikan oleh sisi yang menghubungkan dua simpul tertentu.

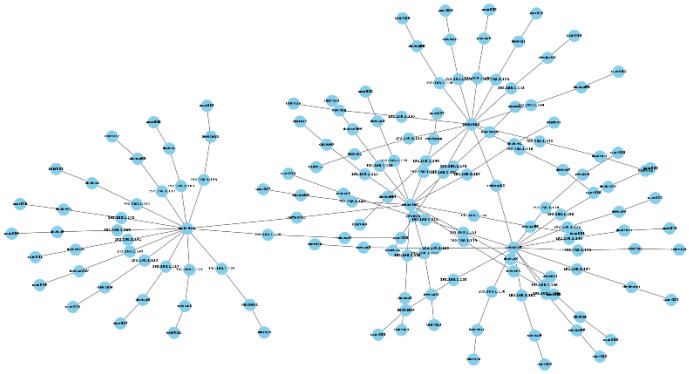
```
import networkx as nx
import matplotlib.pyplot as plt

def load_config(file_path):
    with open(file_path, 'r') as file:
        config_data = file.readlines()
    return [line.strip() for line in
config_data]

def build_graph(file_path):
    G = nx.Graph()

    config_data = load_config(file_path)
    for line in config_data:
        elements = line.split(',')
        for i in range(len(elements)-1):
            G.add_edge(elements[i],
elements[i + 1])

    return G

def visualize_graph(G):
    pos = nx.spring_layout(G)  # Layout graf
(bisa diganti dengan yang lain)
    nx.draw(G, pos, with_labels=True,
font_weight='bold', font_size = 5,
```

```
node_color='skyblue', edge_color='gray')
    plt.show()

def main():
    # path dapat diganti sesuai dengan lokasi
file konfigurasi Anda
    file_path = './config/jejakdigital.txt'

    try:
        graph = build_graph(file_path)
        visualize_graph(graph)
    except FileNotFoundError:
        print(f"File not found:
{file_path}")
    except Exception as e:
        print(f"An error occurred: {str(e)}")

if __name__ == "__main__":
    main()
```

Berdasarkan program python yang telah dibuat, dapat divisualisasikan data ke dalam representasi graf yang menyatakan hubungan dalam data jejak digital tersebut. Jika terdapat simpul yang saling bertetanggaan, berarti data yang diwakilkan oleh simpul tersebut memiliki suatu hubungan. Program ini menerima masukan berdasarkan file konfigurasi external dengan format .txt yang nantinya akan divisualisasikan ke dalam graf.

```
user001,deviceA,192.168.1.101,websiteA,network1
user002,deviceB,192.168.1.102,websiteB,network2
user003,deviceC,192.168.1.103,websiteC,network1
user004,deviceD,192.168.1.104,websiteD,network2
user005,deviceE,192.168.1.105,websiteA,network1
user006,deviceF,192.168.1.106,websiteB,network2
user007,deviceG,192.168.1.107,websiteC,network1
user008,deviceH,192.168.1.108,websiteD,network2
user009,deviceI,192.168.1.109,websiteA,network1
user010,deviceJ,192.168.1.110,websiteB,network2
user011,deviceK,192.168.1.111,websiteC,network1
user012,deviceL,192.168.1.112,websiteD,network2
user013,deviceM,192.168.1.113,websiteA,network1
user014,deviceN,192.168.1.114,websiteB,network2
user015,deviceO,192.168.1.115,websiteC,network1
user016,deviceP,192.168.1.116,websiteD,network2
user017,deviceQ,192.168.1.117,websiteA,network1
user018,deviceR,192.168.1.118,websiteB,network2
user019,deviceS,192.168.1.119,websiteC,network1
user020,deviceT,192.168.1.120,websiteD,network2
user021,deviceU,192.168.1.121,websiteA,network1
user022,deviceV,192.168.1.122,websiteB,network2
user023,deviceW,192.168.1.123,websiteC,network1
user024,deviceX,192.168.1.124,websiteD,network2
user025,deviceY,192.168.1.125,websiteA,network1
user026,deviceZ,192.168.1.126,websiteB,network2
user027,deviceAA,192.168.1.127,websiteC,network1
user028,deviceBB,192.168.1.128,websiteD,network2
user029,deviceCC,192.168.1.129,websiteA,network1
user030,deviceDD,192.168.1.130,websiteB,network2
user031,deviceEE,192.168.1.131,websiteC,network1
user032,deviceFF,192.168.1.132,websiteD,network2
user033,deviceGG,192.168.1.133,websiteA,network1
user034,deviceHH,192.168.1.134,websiteB,network2
user035,deviceII,192.168.1.135,websiteC,network1
user036,deviceJJ,192.168.1.136,websiteD,network2
user037,deviceKK,192.168.1.137,websiteA,network1
user038,deviceLL,192.168.1.138,websiteB,network2
user039,deviceMM,192.168.1.139,websiteC,network1
user040,deviceNN,192.168.1.140,websiteD,network2
user041,deviceOO,192.168.1.141,websiteA,network1
user042,devicePP,192.168.1.142,websiteB,network2
user043,deviceQQ,192.168.1.143,websiteC,network1
user044,deviceRR,192.168.1.144,websiteD,network2
user045,deviceSS,192.168.1.145,websiteA,network1
user046,deviceTT,192.168.1.146,websiteB,network2
user047,deviceUU,192.168.1.147,websiteC,network1
user048,deviceVV,192.168.1.148,websiteD,network2
user049,deviceWW,192.168.1.149,websiteA,network1
user050,deviceXX,192.168.1.150,websiteB,network2
```

Gambar 9. konfigurasi jejakdigital.txt



Gambar 10. Visualisasi jejak digital menggunakan program berdasarkan konfigurasi jejakdigital.txt

Melalui pemodelan data seperti ini, data dapat diinterpretasikan dengan cepat dan efektif. Data yang terdapat dalam visualisasi tersebut dikatakan terhubung apabila simpul yang mewakilkan data tersebut bertetangga atau terhubung secara langsung. Keterkaitan antar data juga dapat dilihat berdasarkan lintasan antar simpul yang mewakili data tersebut. Jika graf memiliki lintasan antara suatu simpul dengan simpul lainnya, maka data yang diwakilkan simpul tersebut saling terhubung.

IV. Visualisasi Jejak Digital dalam Mendeteksi Anomali Keamanan Informasi

Visualisasi jejak digital ke dalam graf dapat diterapkan lebih lanjut untuk mendeteksi anomali keamanan informasi. Anomali pada data dapat dideteksi menggunakan algoritma clustering yang telah dibahas pada Bab II. Algoritma clustering dapat membantu dalam mengidentifikasi pola atau kelompok data normal di dalam sistem. Dengan memahami sesuatu yang dianggap normal, program dapat mendeteksi data atau kejadian yang di luar pola tersebut sebagai anomali. Data yang memiliki jarak diluar batas normal dari pusat kelompok dapat dianggap sebagai *outliers* atau anomali. Dalam kasus ini, pusat kelompok merupakan jaringan atau server dimana pengguna, perangkat, Alamat IP, maupun situs web terhubung dengan jaringan atau server.

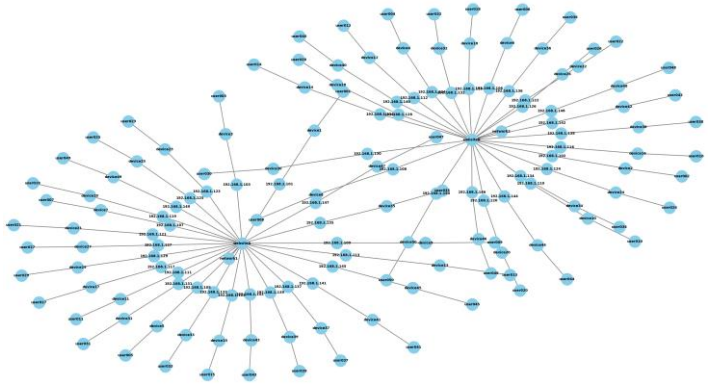
```
import pandas as pd
from sklearn.cluster import KMeans
from sklearn.preprocessing import
StandardScaler, LabelEncoder
from sklearn.cluster import DBSCAN

def load_data(file_path):
    try:
        with open(file_path, 'r') as file:
            data = ...
        return data
    except FileNotFoundError:
        print(f"File not found: {file_path}")
        return None

def preprocess_data(data):
    data_for_clustering = data.drop('user',
axis=1, errors='ignore')

    print("Available columns:",
```

```
data_for_clustering.columns)

    if 'IP' in data_for_clustering.columns:
        data_for_clustering['IP'] =
data_for_clustering['IP'].apply(lambda x:
int(''.join([i.zfill(3) for i in
x.split('.')])))

    label_encoder = LabelEncoder()
    data_for_clustering['device'] =
label_encoder.fit_transform(data_for_clusteri
ng['device'])

    features = data_for_clustering[['device',
'IP']]

    scaler = StandardScaler()
    features_scaled =
scaler.fit_transform(features)

    return features_scaled

def detect_anomalies(data, features_scaled):
    dbscan = DBSCAN(eps=0.5, min_samples=5)
    data['anomaly'] =
dbscan.fit_predict(features_scaled)
    anomalies = data[data['anomaly'] == -1]
    return anomalies

def main():
    file_path = './config/anomalies.csv'
    df = pd.read_csv(file_path)
    # df = pd.DataFrame(data)

    if df is not None:
        features_scaled = preprocess_data(df)

        # Apply KMeans clustering
        kmeans = KMeans(n_clusters=2,
random_state=42)
        df['cluster'] =
kmeans.fit_predict(features_scaled)
        features_scaled = preprocess_data(df)
        anomalies = detect_anomalies(df,
features_scaled)
        # Display the clustered data
        print("Clustered data (Menggunakan
algoritma K-Means) :")
        print(df)

        # Display the anomalies
        if anomalies.empty == False:
            print("\n")
            print("Anomalies(Menggunakan
algoritma DBSCAN) :")
            print(anomalies)
        else:
            print("\n")
        print("Tidak ada anomali yang
terdeteksi pada data jejak digital Anda")

  if __name__ == "__main__":
      main()
```

Berdasarkan program yang telah direalisasikan menggunakan bahasa pemrograman python, dapat dideteksi anomali berdasarkan masukan data pada program. Alur utama dari program ini adalah melakukan pengelompokan data menggunakan algoritma K-means. Setelah dilakukan pengelompokan data program akan melakukan proses deteksi anomali menggunakan algoritma DBSCAN. Program akan mencari outliers dan menampilkan outliers sebagai anomali pada data yang diberikan. Masukan konfigurasi data yang digunakan pada program adalah anomalies.csv. Konfigurasi data anomalies.csv berisi data jejak digital yang memiliki anomali.

```
user,device,IP,website,network
user001,device1,192.168.1.101,websiteA,network1
user002,device2,192.168.1.102,websiteB,network2
user003,device3,192.168.1.103,websiteA,network1
user004,device4,192.168.1.104,websiteB,network2
user005,device5,192.168.1.105,websiteA,network1
user006,device6,192.168.1.106,websiteB,network2
user007,device7,192.168.1.107,websiteA,network1
user008,device8,192.168.1.108,websiteB,network2
user009,device9,192.168.1.109,websiteA,network1
user010,device10,192.168.1.110,websiteB,network2
user011,device11,192.168.1.111,websiteA,network1
user012,device12,192.168.1.112,websiteB,network2
user013,device13,192.168.1.113,websiteA,network1
user014,device14,192.168.1.114,websiteB,network2
user015,device15,192.168.1.115,websiteA,network1
user016,device16,192.168.1.116,websiteB,network2
user017,device17,192.168.1.117,websiteA,network1
user018,device18,192.168.1.118,websiteB,network2
user019,device19,192.168.1.119,websiteA,network1
user020,device20,192.168.1.120,websiteB,network2
user021,device21,192.168.1.121,websiteA,network1
user022,device22,192.168.1.122,websiteB,network2
user023,device23,192.168.1.123,websiteA,network1
user024,device24,192.168.1.124,websiteB,network2
user025,device25,192.168.1.125,websiteA,network1
user026,device26,192.168.1.126,websiteB,network2
user027,device27,192.168.1.127,websiteA,network1
user028,device28,192.168.1.128,websiteB,network2
user029,device29,192.168.1.129,websiteA,network1
user030,device30,192.168.1.130,websiteB,network2
user031,device31,192.168.1.131,websiteA,network1
user032,device32,192.168.1.132,websiteB,network2
user033,device33,192.168.1.133,websiteA,network1
user034,device34,192.168.1.134,websiteB,network2
user035,device35,192.168.1.135,websiteA,network1
user036,device36,192.168.1.136,websiteB,network2
user037,device37,192.168.1.137,websiteA,network1
user038,device38,192.168.1.138,websiteB,network2
```

Gambar 11. Cuplikan konfigurasi anomalies.csv



Gambar 12. Visualisasi jejak digital menggunakan program pada Bab III berdasarkan konfigurasi anomalies.csv

	user	device	IP	website	network	cluster
0	user001	device1	192.168.1.101	websiteA	network1	1
1	user002	device2	192.168.1.102	websiteB	network2	1
2	user003	device3	192.168.1.103	websiteA	network1	1
3	user004	device4	192.168.1.104	websiteB	network2	1
4	user005	device5	192.168.1.105	websiteA	network1	0
5	user006	device6	192.168.1.106	websiteB	network2	0
6	user007	device7	192.168.1.107	websiteA	network1	0
7	user008	device8	192.168.1.108	websiteB	network2	0
8	user009	device9	192.168.1.109	websiteA	network1	0
9	user010	device10	192.168.1.110	websiteB	network2	1
10	user011	device11	192.168.1.111	websiteA	network1	1
11	user012	device12	192.168.1.112	websiteB	network2	1
12	user013	device13	192.168.1.113	websiteA	network1	1
13	user014	device14	192.168.1.114	websiteB	network2	1
14	user015	device15	192.168.1.115	websiteA	network1	1
15	user016	device16	192.168.1.116	websiteB	network2	1
16	user017	device17	192.168.1.117	websiteA	network1	1
17	user018	device18	192.168.1.118	websiteB	network2	1
18	user019	device19	192.168.1.119	websiteA	network1	1
19	user020	device20	192.168.1.120	websiteB	network2	1
20	user021	device21	192.168.1.121	websiteA	network1	1
21	user022	device22	192.168.1.122	websiteB	network2	1
22	user023	device23	192.168.1.123	websiteA	network1	1
23	user024	device24	192.168.1.124	websiteB	network2	1
24	user025	device25	192.168.1.125	websiteA	network1	1
25	user026	device26	192.168.1.126	websiteB	network2	1
26	user027	device27	192.168.1.127	websiteA	network1	1
27	user028	device28	192.168.1.128	websiteB	network2	1
28	user029	device29	192.168.1.129	websiteA	network1	0
29	user030	device30	192.168.1.130	websiteB	network2	0
30	user031	device31	192.168.1.131	websiteA	network1	0
31	user032	device32	192.168.1.132	websiteB	network2	0
32	user033	device33	192.168.1.133	websiteA	network1	0
33	user034	device34	192.168.1.134	websiteB	network2	0
34	user035	device35	192.168.1.135	websiteA	network1	0
35	user036	device36	192.168.1.136	websiteB	network2	0
36	user037	device37	192.168.1.137	websiteA	network1	0
37	user038	device38	192.168.1.138	websiteB	network2	0
38	user039	device39	192.168.1.139	websiteA	network1	0
39	user040	device40	192.168.1.140	websiteB	network2	0
40	user041	device41	192.168.1.141	websiteA	network1	0
41	user042	device42	192.168.1.142	websiteB	network2	0
42	user043	device43	192.168.1.143	websiteA	network1	0
43	user044	device44	192.168.1.144	websiteB	network2	0
44	user045	device45	192.168.1.145	websiteA	network1	0
45	user046	device46	192.168.1.146	websiteB	network2	0
46	user047	device47	192.168.1.147	websiteA	network1	0
47	user048	device48	192.168.1.148	websiteB	network2	0
48	user049	device49	192.168.1.149	websiteA	network1	0
49	user050	device50	192.168.1.150	websiteB	network2	0

Gambar 13. Hasil pengelompokan menggunakan K-means

Anomalies (Menggunakan algoritma DBSCAN) :

	user	device	IP	website	network	cluster	anomaly
0	user001	device1	192.168.1.101	websiteA	network1	1	-1
1	user002	device2	192.168.1.102	websiteB	network2	1	-1
2	user003	device3	192.168.1.103	websiteA	network1	1	-1
3	user004	device4	192.168.1.104	websiteB	network2	1	-1

Gambar 14. Hasil analisis anomali menggunakan DBSCAN

Berdasarkan hasil yang diberikan oleh program, mendeteksi anomali pada data yang telah divisualisasikan menjadi graf (gambar 12) sulit dilakukan dengan kapasitas visual manusia. Diperlukan tahapan lebih lanjut untuk mendeteksi anomali pada graf jejak digital seperti penerapan algoritma clustering. Anomali dapat terdeteksi setelah melalui proses pengelompokan menggunakan K-means seperti pada gambar 13 dan pencarian titik data outliers menggunakan DBSCAN pada gambar 14.

V. Kesimpulan

Graf dapat diimplementasikan ke dalam bidang keamanan informasi. Diperlukan proses lebih lanjut setelah melakukan visualisasi data ke dalam representasi graf. Algoritma clustering diperlukan dalam mendeteksi anomali pada data jejak digital dikarenakan kemampuan visual manusia yang terbatas dalam mendeteksi anomali pada graf dengan volume data yang besar. Pemanfaatan graf masih dapat terus dikembangkan dalam keamanan informasi dikarenakan graf memiliki beberapa manfaat yang sangat mangkus dan sangkil di dunia informatika.

VI. Ucapan Terima Kasih

Penulis ingin memanjatkan syukur atas berkat dan rahmat yang telah diberikan Tuhan Yang Maha Esa, sehingga bisa menyelesaikan karya tulis berjudul “Aplikasi Graf dalam Menganalisis Jejak Digital untuk Mendeteksi Anomali pada Keamanan Informasi”.

Selain itu, Penulis juga ingin mengucapkan terima kasih kepada pihak keluarga yang telah memberikan semangat dan dukungan dalam menyelesaikan karya tulis ini. Ucapan terima kasih Penulis sampaikan juga kepada Dr. Ir. Rinaldi Munir, M.T. dan Monterico Adrian, S.T., M.T. selaku dosen pengampu Mata Kuliah Matematika Diskrit ITB untuk Kelas 3, atas segala pedoman dan ilmu yang telah diajarkan.

Daftar Pustaka

[1] Madden, M., Fox, S., & Vitak, J. (16 Desember 2007). Digital footprints. Pew Research Center: Internet, Science & Tech. https://www.pewresearch.org/internet/2007/12/16/digital-footprints/ (diakses 5 Desember 2023, pukul 12.00 WIB).

[2] Enzo. (14 Juni 2022). Graphs for cybersecurity: Introduction. Graph Database & Analytics. https://neo4j.com/blog/graphs-for-cybersecurity/ (diakses 3 Desember 2023, pukul 16.00 WIB).

[3] Kaspersky. (17 Agustus 2023). What is cyber security?. www.kaspersky.com. https://www.kaspersky.com/resource-center/definitions/what-is-cyber-security (diakses 2 Desember 2023, pukul 18.00 WIB).

[4] Kaushi, S. Clustering | Introduction, Different Methods, and Applications. Analytics Vidhya. https://www.analyticsvidhya.com/blog/2016/11/an-introduction-to-clustering-and-different-methods-of-clustering/ (diakses 8 Desember 2023, pukul 10.00 WIB).

[5] Munir, Rinaldi. (2020). “Graf: Bagian 1”. https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2023-2024/19-Graf-Bagian1-2023.pdf (diakses 30 November 2023, pukul 13.00 WIB).

[6] Munir, Rinaldi. (2020). “Graf: Bagian 3”. https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2023-2024/21-Graf-Bagian3-2023.pdf (diakses 1 Desember 2023, pukul 19.00 WIB).

[7] Verizon Ventures. What is a digital footprint?. verizon. https://www.verizon.com/about/blog/digital-footprint-definition-examples-and-ways-reduce (diakses 3 Desember 2023, pukul 20.00 WIB).

Pernyataan

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 8 Desember 2023

Aland Mulia Pratama 13522124