**Міністерство освіти і науки України**

**Чернівецький національний університет**

**імені Юрія Федьковича**

**Інститут фізико-технічних і комп’ютерних наук**

(повна назва факультету)

**Кафедра кореляційної оптики**

(повна назва кафедри)

**4 частина курсової роботи**

з теми “Аналіз статистики використання мобільних мереж”

З Інформатики .

(назва дисципліни)

Виконала: Студентка 1 курсу, групи 124

Спеціальності

172 «Телекомунікації та радіотехніка»

(Телекомунікації)

(шифр і назва спеціальності)

Продан Д.В.

(прізвище та ініціали, підпис)

Керівник Рябий П.А.

(прізвище та ініціали, підпис)

Оцінка: Національна шкала

Кількість балів: Оцінка: ЄКТС

Чернівці 2025

**Зміст**

[**Вступ 3**](#_Toc198810258)

[**Структура проекту 4**](#_Toc198810259)

[**1. Завантаження та обробка даних 4**](#_Toc198810260)

[**2. Інтерактивна фільтрація даних 5**](#_Toc198810261)

[**3. Візуалізація та аналіз даних 5**](#_Toc198810262)

[**4. Виявлення та аналіз аномалій 6**](#_Toc198810263)

## Вступ

Цей проєкт надає інтерактивне веб-середовище для аналізу та візуалізації мережевого трафіку з можливістю виявлення аномалій. Використовуючи Streamlit, проєкт дозволяє аналітикам та фахівцям з кібербезпеки досліджувати великі обсяги мережевих даних, виявляти закономірності та потенційні загрози

## Структура проекту

dashboard.py - Основний файл веб-інтерфейсу

dataset.py - Скрипт для генерації/обробки даних трафіку

src/

├── data\_loader.py - Завантаження наборів даних

├── data\_cleaner.py - Очищення та попередня обробка даних

└── config.py - Конфігураційні параметри

data/

├── dataset.csv - Датасет мережевого трафіку

└── dataset1.csv - Альтернативний датасет

Основні функціональні можливості

### **1. Завантаження та обробка даних**

Dashboard підтримує роботу з двома типами наборів даних: синтетичними та реальними. Дані завантажуються за допомогою спеціальної функції load\_dataset() з модуля src.data\_loader:

# Завантаження наборів даних

df\_real = load\_dataset('real')

df\_synthetic = load\_dataset()

# Очищення даних

df = clean\_data(df, NUMERIC\_COLUMNS)

Користувач може перемикатися між наборами даних через інтерфейс:

dataset\_option = st.sidebar.selectbox(

"Оберіть набір даних:",

options=["Синтетичні дані", "Реальні дані"]

)

### **2. Інтерактивна фільтрація даних**

Dashboard надає різноманітні можливості фільтрації для детального аналізу:

# Фільтрація за протоколом

protocols = ['Всі'] + sorted(df['proto'].unique().tolist())

selected\_protocol = st.sidebar.selectbox("Протокол:", protocols)

# Фільтрація за діапазоном дат

selected\_date\_range = st.sidebar.date\_input(

"Діапазон дат:",

value=(date\_min, date\_max),

min\_value=date\_min,

max\_value=date\_max

)

# Застосування фільтрів аномалій

if 'anomaly' in filtered\_df.columns and anomaly\_filter != 'Всі дані':

if anomaly\_filter == 'Тільки нормальні':

filtered\_df = filtered\_df[filtered\_df['anomaly'] == 0]

else: # 'Тільки аномалії'

filtered\_df = filtered\_df[filtered\_df['anomaly'] == 1]

### **3. Візуалізація та аналіз даних**

Дашборд організований у вигляді вкладок, кожна з яких відповідає за певний тип аналізу:

1. Розподіли числових показників

fig = px.histogram(filtered\_df, x=column, nbins=50, marginal="box",

color='anomaly' if 'anomaly' in filtered\_df.columns else None,

color\_discrete\_map={0: 'blue', 1: 'red'})

1. Аналіз протоколів і сервісів

proto\_counts = filtered\_df['proto'].value\_counts()

fig = px.pie(values=proto\_counts.values, names=proto\_counts.index,

title="Розподіл протоколів")

1. Часовий аналіз

hourly\_traffic = filtered\_df.groupby('hour')['sbytes'].sum().reset\_index()

fig = px.line(hourly\_traffic, x='hour', y='sbytes',

labels={'hour': 'Година доби', 'sbytes': 'Обсяг даних'},

title="Розподіл трафіку за годинами доби")

1. Географічна візуалізація трафіку

fig = px.choropleth(country\_traffic,

locations="iso\_alpha",

color="total\_bytes",

hover\_name=country\_col,

title=f"Розподіл трафіку за країнами ({traffic\_direction.lower()})",

color\_continuous\_scale=px.colors.sequential.Plasma)

### **4. Виявлення та аналіз аномалій**

Dashboard включає функціональність для автоматичного виявлення аномалій у мережевому трафіку:

# Виявлення DDoS атак

ddos\_mask = (df['spkts'] / df['dur'] > ddos\_threshold\_pkt\_rate) & (df['dur'] < 0.1)

df.loc[ddos\_mask, 'anomaly'] = 1

df.loc[ddos\_mask, 'anomaly\_type'] = 'ddos'

# Виявлення неправильних конфігурацій

wrong\_ttl\_mask = ((df['sttl'] <= 2) | (df['sttl'] >= 254) |

(df['dttl'] <= 2) | (df['dttl'] >= 254))

df.loc[wrong\_ttl\_mask, 'anomaly'] = 1

df.loc[wrong\_ttl\_mask, 'anomaly\_type'] = 'misconfig:wrong\_ttl'

Після виявлення аномалій можна провести порівняльний аналіз між нормальним та аномальним трафіком:

# Порівняння показників нормального і аномального трафіку

fig = go.Figure()

fig.add\_trace(go.Box(y=df[df['anomaly'] == 0][comparison\_metric],

name='Нормальний трафік',

marker\_color='blue'))

fig.add\_trace(go.Box(y=anomaly\_data[comparison\_metric],

name='Аномальний трафік',

marker\_color='red'))