Przetwarzanie Strumieni Danych
Zajecia Zintegrowane 1
Filip Pawłowski 310867

Uruchomiono maszynę wirtualną, w odpowiednich katalogach przekopiowano i wypakowano pliki źródłowe Kafki, Kafdrop'a oraz Flink'a. Następnie stworzono pliki konfiguracyjne: /etc/systemd/system/kafka.service oraz /etc/systemd/system/zookeper.servivce. Przy pomocy narzędzia *nano* zostały one wypełnione ustawieniami podanymi w instrukcji do laboratorium.

Tak ustawioną Kafkę uruchomiono przy pomocy polecenia: **sudo systemctl start kafka.**
O sukcesie świadczy rezultat sprawdzenia statusu przy pomocy **sudo systemctl status kafka:**

```
kafka@ubuntu3:~/kafka$ sudo systemctl status kafka
* kafka.service
     Loaded: loaded (/etc/systemd/system/kafka.service; disabled; vendor preset: enabled)
     Active: active (running) since Thu 2025-03-27 14:46:05 CET; 49s ago
   Main PID: 5891 (sh)
      Tasks: 73 (limit: 9437)
     Memory: 337.0M
        CPU: 12.594s
     CGroup: /system.slice/kafka.service
             |-5891 /bin/sh -c "/home/kafka/kafka/bin/kafka-server-start.sh /home/kafka/kafk>
             `-5893 java -Xmx1G -Xms1G -server -XX:+UseG1GC -XX:MaxGCPauseMillis=20 -XX:Init>

mar 27 14:46:05 ubuntu3 systemd[1]: Started kafka.service.
```

Stworzony został topic kafki o nazwie „PSDTopic" - wykorzystywany w dalszych etapach laboratorium.

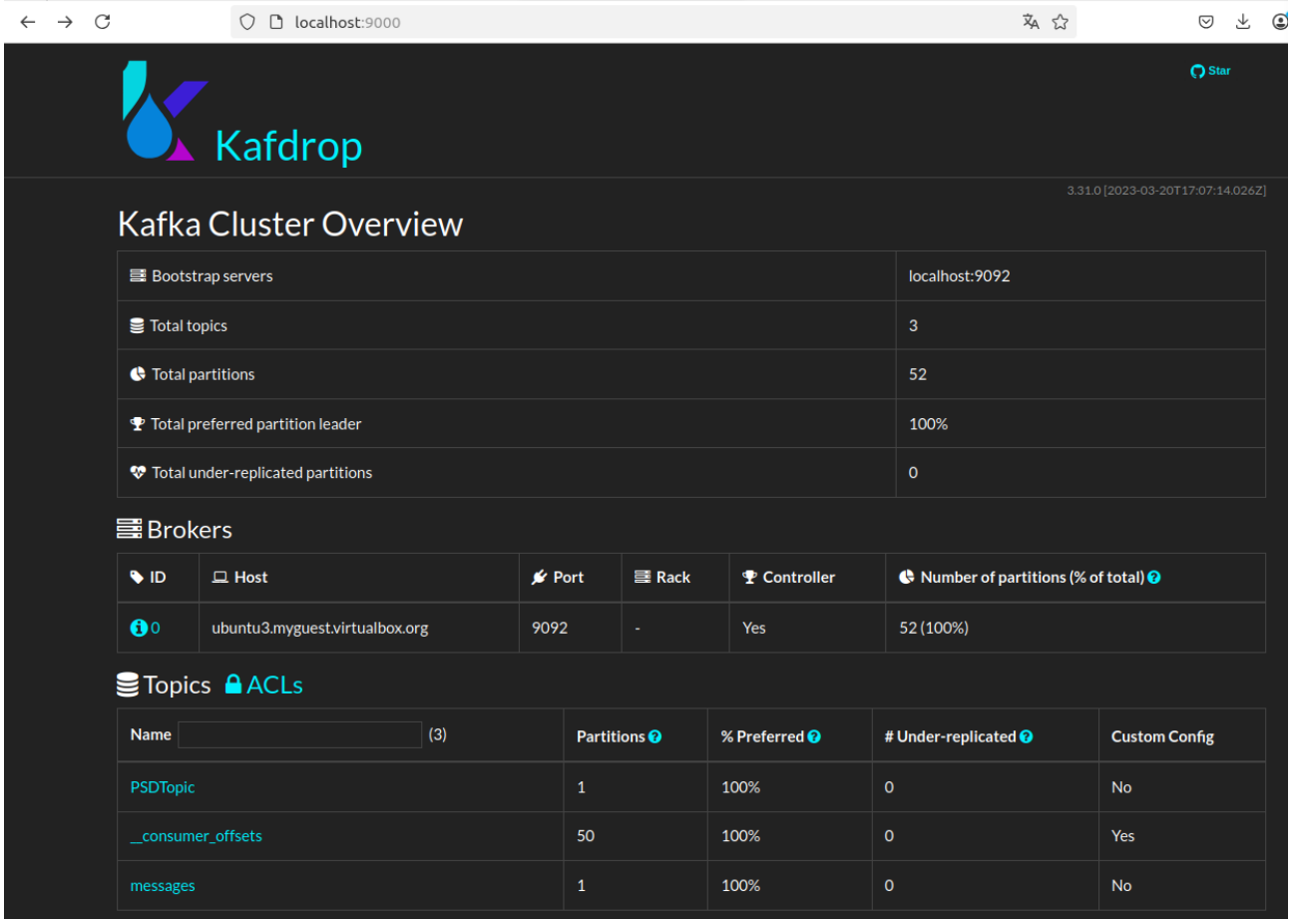
```
kafka@ubuntu3:~/kafka$ sudo ~kafka/kafka/bin/kafka-topics.sh --create --bootstrap-server localhost:90
92 --replication-factor 1 --partitions 1 --topic PSDTopic
Created topic PSDTopic.
```

Jego funkcjonowanie zweryfikowano przy pomocy skryptów **kafka-console-producer.sh** oraz **kafka-console-consumer.sh**:

```
kafka@ubuntu3:~/kafka$ echo "Hello, World223" | ~kafka/kafka/bin/kafka-console-producer.sh --broker-l
ist localhost:9092 --topic PSDTopic > /dev/null
kafka@ubuntu3:~/kafka$ ~kafka/kafka/bin/kafka-console-consumer.sh --bootstrap-server localhost:9092 -
-topic PSDTopic --from-beginning
Hello, World
Hello, World22
Hello, World223
Hello, World223
^CProcessed a total of 4 messages
```

Uruchomiony został plik jar zawierajacy obraz programu Kafrop, co można zobaczyc tez na screenie ponizej:

```
psd@ubuntu3:~$ sudo java --add-opens=java.base/sun.nio.ch=ALL-UNNAMED -jar ~kafka/Downloads/kafdrop-3.31.0.jar --kafka.brokerConnect=local
host:9092
[sudo] password for psd:
2025-03-27 15:44:43.790  INFO 14970 [kground-preinit] o.h.v.i.u.Version                        : HV000001: Hibernate Validator 6.2.5.Final
2025-03-27 15:44:43.879  INFO 14970 [           main] o.s.b.StartupInfoLogger                 : Starting Kafdrop v3.31.0 using Java 11.0.
26 on ubuntu3 with PID 14970 (/home/kafka/Downloads/kafdrop-3.31.0.jar started by root in /home/psd)
2025-03-27 15:44:43.893  INFO 14970 [           main] o.s.b.SpringApplication                 : No active profile set, falling back to 1
default profile: "default"
2025-03-27 15:44:48.248  INFO 14970 [           main] i.u.s.s.ServletContextImpl              : Initializing Spring embedded WebApplicati
onContext
2025-03-27 15:44:48.249  INFO 14970 [           main] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initializatio
n completed in 4226 ms
2025-03-27 15:44:48.817  INFO 14970 [           main] k.c.KafkaConfiguration                  : Checking truststore file kafka.truststore
.jks
2025-03-27 15:44:48.819  INFO 14970 [           main] k.c.KafkaConfiguration                  : Checking keystore file kafka.keystore.jks
2025-03-27 15:44:48.819  INFO 14970 [           main] k.c.KafkaConfiguration                  : Checking properties file kafka.properties
2025-03-27 15:44:49.057  INFO 14970 [           main] k.c.KafkaConfiguration                  : Checking truststore file kafka.truststore
.jks
2025-03-27 15:44:49.058  INFO 14970 [           main] k.c.KafkaConfiguration                  : Checking keystore file kafka.keystore.jks
2025-03-27 15:44:49.058  INFO 14970 [           main] k.c.KafkaConfiguration                  : Checking properties file kafka.properties
2025-03-27 15:44:49.170  INFO 14970 [           main] k.s.BuildInfo                           : Kafdrop version: 3.31.0, build time: 2023
-03-20T17:07:14.026Z
2025-03-27 15:44:50.526  INFO 14970 [           main] o.s.b.a.e.w.EndpointLinksResolver       : Exposing 13 endpoint(s) beneath base path
 '/actuator'
2025-03-27 15:44:52.088  INFO 14970 [           main] i.u.Undertow                            : starting server: Undertow - 2.2.20.Final
2025-03-27 15:44:52.114  INFO 14970 [           main] o.x.Xnio                                : XNIO version 3.8.7.Final
2025-03-27 15:44:52.151  INFO 14970 [           main] o.x.n.NioXnio                           : XNIO NIO Implementation Version 3.8.7.Fin
al
2025-03-27 15:44:52.238  INFO 14970 [           main] o.j.t.Version                           : JBoss Threads version 3.1.0.Final
2025-03-27 15:44:52.371  INFO 14970 [           main] o.s.b.w.e.u.UndertowWebServer           : Undertow started on port(s) 9000 (http)
2025-03-27 15:44:53.144  INFO 14970 [           main] o.s.b.StartupInfoLogger                 : Started Kafdrop in 10.314 seconds (JVM ru
nning for 12.688)
```



Dodatkowo przetestowano wymiane informacji przy pomocy skrytpów jezyka Python, symulujacych detekcje ruchu przez sensory.

producent.py:

producer.py

```python
import random
import string

detector_ids = list(range(1, 101))

def generate_message1() -> dict:
    detector_id = random.choice(detector_ids)

    # Generate a random message
    temp=f"Movement detected at field: A{random.randint(0,30)}"
    message = ''.join(temp)
    return {
    'detector_id': detector_id,
    'message': message
    }

import time
import json
import random
from datetime import datetime
from kafka import KafkaProducer

# Messages will be serialized as JSON
def serializer(message):
    return json.dumps(message).encode('utf-8')

# Kafka Producer
producer = KafkaProducer(
    bootstrap_servers=['localhost:9092'],
    value_serializer=serializer,
)

if __name__ == '__main__':
    # Infinite loop - runs until you kill the program
    while True:
        # Generate a message
        dummy_message = generate_message1()
        # Send it to our 'messages' topic
        print(f'Producing message @ {datetime.now()} | Message = {str(dummy_message)}')

        producer.send('PSDTopic', dummy_message)

        # Sleep for a random number of seconds
        time_to_sleep = random.randint(1, 11)
        time.sleep(time_to_sleep)
```

consumer.py:

consumer.py

```python
from kafka import KafkaConsumer

if __name__ == '__main__':
    # Kafka Consumer
    consumer = KafkaConsumer(
    'PSDTopic',
    bootstrap_servers='localhost:9092',
    auto_offset_reset='earliest',
    )
for message in consumer:
    try:
        print(json.loads(message.value))
    except json.decoder.JSONDecodeError:
        print(f"Raw value: {message.value}")
```

Po uruchomieniu obu skryptow, wiadomosci są generowane przez producenta i konsumowane przez konsumenta.

```
psd@ubuntu3:~/python-kafka$ python3 producer.py
Producing message @ 2025-03-27 15:34:56.346569 | Message = {'detector_id': 12, 'message': 'Movement detected at field: A3'}
Producing message @ 2025-03-27 15:34:59.355891 | Message = {'detector_id': 97, 'message': 'Movement detected at field: A13'}
Producing message @ 2025-03-27 15:35:07.363736 | Message = {'detector_id': 16, 'message': 'Movement detected at field: A19'}
Producing message @ 2025-03-27 15:35:15.372215 | Message = {'detector_id': 6, 'message': 'Movement detected at field: A22'}
Producing message @ 2025-03-27 15:35:18.375771 | Message = {'detector_id': 65, 'message': 'Movement detected at field: A17'}
Producing message @ 2025-03-27 15:35:21.378737 | Message = {'detector_id': 9, 'message': 'Movement detected at field: A19'}
Producing message @ 2025-03-27 15:35:27.384785 | Message = {'detector_id': 96, 'message': 'Movement detected at field: A17'}
Producing message @ 2025-03-27 15:35:32.392944 | Message = {'detector_id': 17, 'message': 'Movement detected at field: A28'}
Producing message @ 2025-03-27 15:35:40.402817 | Message = {'detector_id': 40, 'message': 'Movement detected at field: A20'}
Producing message @ 2025-03-27 15:35:51.425247 | Message = {'detector_id': 12, 'message': 'Movement detected at field: A13'}
Producing message @ 2025-03-27 15:36:00.435649 | Message = {'detector_id': 90, 'message': 'Movement detected at field: A21'}
Producing message @ 2025-03-27 15:36:04.440302 | Message = {'detector_id': 97, 'message': 'Movement detected at field: A17'}
Producing message @ 2025-03-27 15:36:15.459062 | Message = {'detector_id': 18, 'message': 'Movement detected at field: A15'}
Producing message @ 2025-03-27 15:36:19.462912 | Message = {'detector_id': 69, 'message': 'Movement detected at field: A6'}
Producing message @ 2025-03-27 15:36:25.469207 | Message = {'detector_id': 78, 'message': 'Movement detected at field: A20'}
Producing message @ 2025-03-27 15:36:30.475718 | Message = {'detector_id': 51, 'message': 'Movement detected at field: A9'}
Producing message @ 2025-03-27 15:36:38.506462 | Message = {'detector_id': 36, 'message': 'Movement detected at field: A6'}
Producing message @ 2025-03-27 15:36:42.511906 | Message = {'detector_id': 79, 'message': 'Movement detected at field: A1'}
Producing message @ 2025-03-27 15:36:51.521968 | Message = {'detector_id': 100, 'message': 'Movement detected at field: A2'}
Producing message @ 2025-03-27 15:37:01.540715 | Message = {'detector_id': 97, 'message': 'Movement detected at field: A24'}
Producing message @ 2025-03-27 15:37:11.554436 | Message = {'detector_id': 59, 'message': 'Movement detected at field: A22'}
Producing message @ 2025-03-27 15:37:21.576610 | Message = {'detector_id': 23, 'message': 'Movement detected at field: A9'}
Producing message @ 2025-03-27 15:37:28.582228 | Message = {'detector_id': 4, 'message': 'Movement detected at field: A22'}
Producing message @ 2025-03-27 15:37:31.599984 | Message = {'detector_id': 72, 'message': 'Movement detected at field: A14'}
Producing message @ 2025-03-27 15:37:38.624511 | Message = {'detector_id': 31, 'message': 'Movement detected at field: A9'}
Producing message @ 2025-03-27 15:37:43.631189 | Message = {'detector_id': 14, 'message': 'Movement detected at field: A26'}
Producing message @ 2025-03-27 15:37:52.652295 | Message = {'detector_id': 4, 'message': 'Movement detected at field: A3'}
Producing message @ 2025-03-27 15:37:55.674389 | Message = {'detector_id': 74, 'message': 'Movement detected at field: A12'}
Producing message @ 2025-03-27 15:38:05.692676 | Message = {'detector_id': 88, 'message': 'Movement detected at field: A11'}
Producing message @ 2025-03-27 15:38:07.696738 | Message = {'detector_id': 36, 'message': 'Movement detected at field: A17'}
Producing message @ 2025-03-27 15:38:11.705210 | Message = {'detector_id': 59, 'message': 'Movement detected at field: A0'}
Producing message @ 2025-03-27 15:38:17.710456 | Message = {'detector_id': 94, 'message': 'Movement detected at field: A19'}
Producing message @ 2025-03-27 15:38:20.714088 | Message = {'detector_id': 74, 'message': 'Movement detected at field: A8'}
```

```
psd@ubuntu3:~/python-kafka$ python3 consumer.py
Raw value: b'Hello, World'
Raw value: b'Hello, World22'
Raw value: b'Hello, World223'
Raw value: b'Hello, World223'
{'detector_id': 12, 'message': 'Movement detected at field: A3'}
{'detector_id': 97, 'message': 'Movement detected at field: A13'}
{'detector_id': 16, 'message': 'Movement detected at field: A19'}
{'detector_id': 6, 'message': 'Movement detected at field: A22'}
{'detector_id': 65, 'message': 'Movement detected at field: A17'}
{'detector_id': 9, 'message': 'Movement detected at field: A19'}
{'detector_id': 96, 'message': 'Movement detected at field: A17'}
{'detector_id': 17, 'message': 'Movement detected at field: A28'}
{'detector_id': 40, 'message': 'Movement detected at field: A20'}
{'detector_id': 12, 'message': 'Movement detected at field: A13'}
{'detector_id': 90, 'message': 'Movement detected at field: A21'}
{'detector_id': 97, 'message': 'Movement detected at field: A17'}
{'detector_id': 18, 'message': 'Movement detected at field: A15'}
{'detector_id': 69, 'message': 'Movement detected at field: A6'}
{'detector_id': 78, 'message': 'Movement detected at field: A20'}
{'detector_id': 51, 'message': 'Movement detected at field: A9'}
{'detector_id': 36, 'message': 'Movement detected at field: A6'}
{'detector_id': 79, 'message': 'Movement detected at field: A1'}
{'detector_id': 100, 'message': 'Movement detected at field: A2'}
{'detector_id': 97, 'message': 'Movement detected at field: A24'}
{'detector_id': 59, 'message': 'Movement detected at field: A22'}
{'detector_id': 23, 'message': 'Movement detected at field: A9'}
{'detector_id': 4, 'message': 'Movement detected at field: A22'}
{'detector_id': 72, 'message': 'Movement detected at field: A14'}
{'detector_id': 31, 'message': 'Movement detected at field: A9'}
{'detector_id': 14, 'message': 'Movement detected at field: A26'}
{'detector_id': 4, 'message': 'Movement detected at field: A3'}
{'detector_id': 74, 'message': 'Movement detected at field: A12'}
{'detector_id': 88, 'message': 'Movement detected at field: A11'}
{'detector_id': 36, 'message': 'Movement detected at field: A17'}
{'detector_id': 59, 'message': 'Movement detected at field: A0'}
{'detector_id': 94, 'message': 'Movement detected at field: A19'}
{'detector_id': 74, 'message': 'Movement detected at field: A8'}
```

Poniewaz w watku sa wiadomosci testowe wyslane tez wczesniej przy uzyciu skryptow shell, nalezalo dodac rowniez obsluge dla wiadomosci niezgodnych z formatem JSON.

Wymiane wiadomosci mozna przesledzic rowniez w Kafdrop: