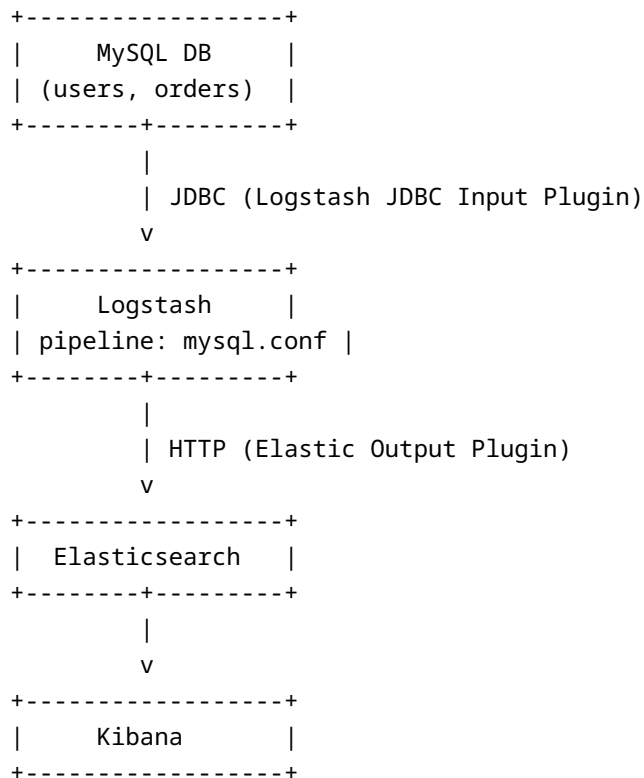




Đồng Bộ MySQL → Elasticsearch Qua Logstash (Docker)



1. Mô hình hoạt động tổng thể



2. Cấu hình `docker-compose.yml`

```

version: '3.8'

services:
  elasticsearch:
    image: docker.elastic.co/elasticsearch/elasticsearch:8.15.0
    container_name: elasticsearch
    environment:
      - discovery.type=single-node
      - xpack.security.enabled=false
      - ES_JAVA_OPTS=-Xms1g -Xmx1g
    ports:
      - "9200:9200"
      - "9300:9300"
    network_mode: bridge

  kibana:
    image: docker.elastic.co/kibana/kibana:8.15.0
    container_name: kibana

```

```

environment:
  - ELASTICSEARCH_HOSTS=http://172.17.0.2:9200 # IP container
Elasticsearch
ports:
  - "5601:5601"
depends_on:
  - elasticsearch
network_mode: bridge

logstash:
image: docker.elastic.co/logstash/logstash:8.15.0
container_name: logstash
environment:
  - LS_JAVA_OPTS=-Xmx1g -Xms1g
volumes:
  - ./logstash-jars:/usr/share/logstash/logstash-jars:ro
  - ./pipeline:/usr/share/logstash/pipeline:ro
ports:
  - "5044:5044"
  - "9600:9600"
depends_on:
  - elasticsearch
network_mode: bridge

```

3. Cấu hình Logstash pipeline (pipeline/mysql.conf)

```

input {
  jdbc {
    jdbc_connection_string => "jdbc:mysql://172.17.0.4:3306/testdb" # IP
MySQL container
    jdbc_user => "root"
    jdbc_password => "root"
    jdbc_driver_library => "/usr/share/logstash/logstash-jars/mysql-
connector-j-8.0.33.jar"
    jdbc_driver_class => "com.mysql.cj.jdbc.Driver"
    schedule => "* * * * *" # chạy mỗi phút
    statement => "SELECT * FROM users" # lấy toàn bộ dữ liệu bảng users
  }
}

output {
  elasticsearch {
    hosts => ["http://172.17.0.2:9200"] # Elasticsearch container
    index => "mysql_index"
  }
  stdout { codec => json_lines }
}

```

📁 Lưu ý: đặt file driver `mysql-connector-j-8.0.33.jar` trong thư mục `logstash-jars/` cùng cấp với `docker-compose.yml`

📦 4. Khởi chạy toàn bộ stack

```
docker-compose up -d
```

- Truy cập Elasticsearch: `http://localhost:9200`
- Truy cập Kibana: `http://localhost:5601`

⚠️ 5. Kiểm tra đồng bộ dữ liệu

Sau khi Logstash chạy, dữ liệu từ bảng `users` trong MySQL sẽ được đẩy sang Elasticsearch.

Kiểm tra:

```
curl -X GET "http://localhost:9200/mysql_index/_search?pretty"
```

Nếu thấy dữ liệu user xuất hiện → kết nối thành công 🎉

🔍 6. Xem dữ liệu trong Kibana

1. Mở Kibana → **Management** → **Stack Management** → **Data Views**
2. Tạo Data View: `mysql_index*`
3. Mở tab **Discover** để xem dữ liệu user
4. Có thể tạo visualization (pie chart, bar chart) theo các field như `age`, `gender`, `country`, v.v.


✅ 7. Kết nối MySQL ↔ Logstash (giải thích)

- **Logstash JDBC Input Plugin** kết nối trực tiếp đến MySQL qua `jdbc_connection_string`
- Mỗi lần chạy (theo `schedule`), Logstash gửi câu lệnh SQL → lấy kết quả → chuyển đổi sang JSON
- Sau đó gửi qua **Elastic Output Plugin** đến Elasticsearch bằng HTTP API

Luồng hoạt động:

```
MySQL (172.17.0.4:3306)
|
▼
Logstash JDBC Plugin → Elasticsearch REST API (172.17.0.2:9200)
|
```

▼
Kibana (localhost:5601)

 **Kết quả:** - Dữ liệu từ bảng `users` trong MySQL được đồng bộ định kỳ mỗi phút sang Elasticsearch - Kibana trực quan hóa dữ liệu dễ dàng qua biểu đồ và dashboard.