# Malware Trends

## 악성코드 수집 및 자동 시각화 시스템

장한길
gksrlf0718@outlook.kr

# Motivation & Objective

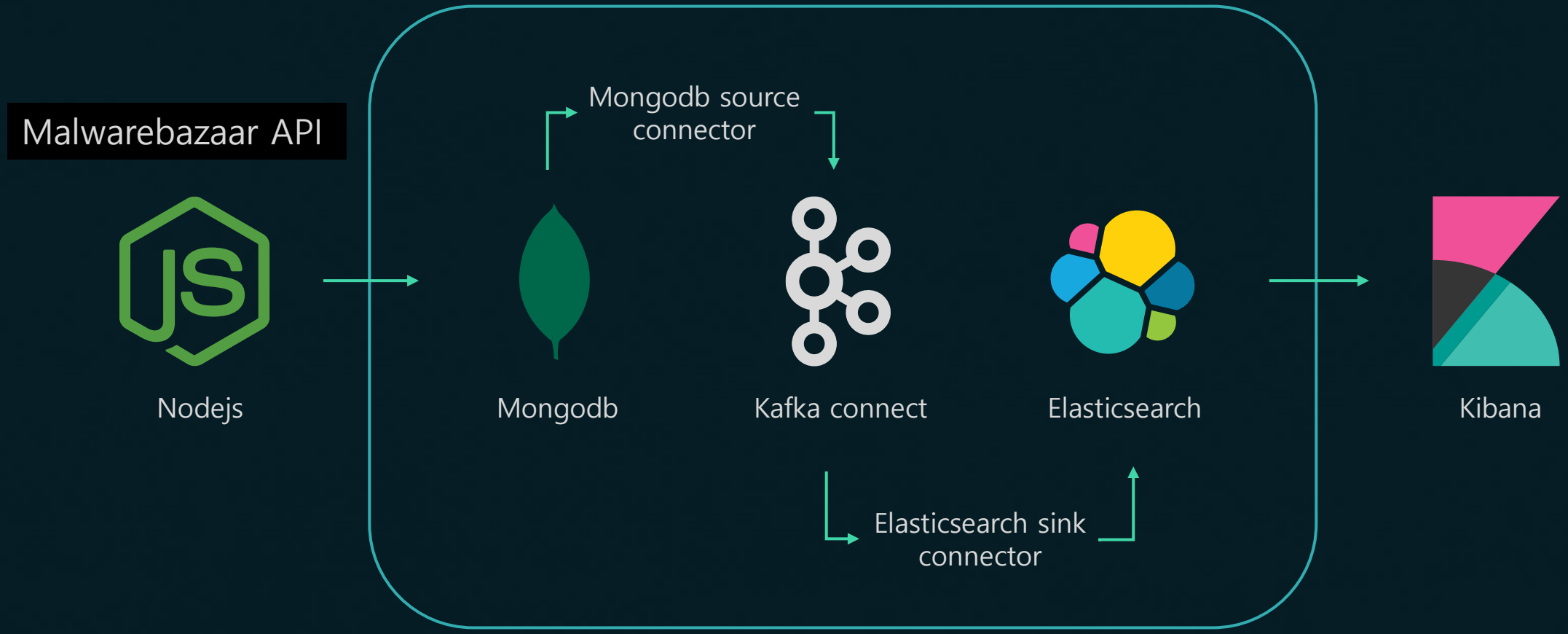
- 악성코드의 메타데이터를 수집, 가공하여 인사이트를 제공하기 위한 파이프라인 구성
  >> 자동화된 파이프라인 필요

- 실시간 데이터 연동 처리와 시각적 위협 분석 리포트 구성

- 기술 스택의 유연한 활용
  >> JSON데이터를 보다 쉽게 다루기 위해 MongoDB를 선택했으나, ELK의 Logstash가 MongoDB를 지원하지 않아, Logstash 대신 Kafka Connect를 이용해 데이터 연동 처리를 구성하였음

  (MySQL도 JSON타입 데이터를 저장할 수 있지만, 문제에 직면하기 위해 MySQL을 사용하지 않음)

# System Architecture



# Data Collection

- 수집 대상 : malwarebazaar API
- 수집 항목 : 날짜, 파일타입, 시그니처, 태그, 국가, 다운로드 수
- 처리 로직 : API 호출 -> JSON 파싱, 데이터 정제 -> MongoDB 저장

```
{ index: { _index: 'malware-kafka.record' } },
{ date: '2025-06-03', country: 'DE', value: 1 },
{ index: { _index: 'malware-kafka.record' } },
{ date: '2025-06-03', country: 'NL', value: 2 },
{ index: { _index: 'malware-kafka.record' } },
{ date: '2025-06-03', filetype_download: 'exe', download: 387 },
{ index: { _index: 'malware-kafka.record' } },
{ date: '2025-06-03', filetype_download: 'exe', download: 304 },
{ index: { _index: 'malware-kafka.record' } },
```

Elasticsearch에서 데이터를 한번에 보낼 때 사용하는 Bulk api형식으로 데이터를 가공, 저장

# Automated Data Integration & Transfer

## MongoDB -> Kafka -> Elasticsearch

```
┌──(root㉿DESKTOP-AUK1FA8)-[~/Kafka/confluent-7.1.2/connector/config]
└─# ls
elasticsearch-sink.properties  http.p12  mongodb-source.properties
```

```
name=elasticsearch-sink
connector.class=io.confluent.connect.elasticsearch.ElasticsearchSinkConnector
tasks.max=1
connection.url=https://localhost:9200

topics=malware-kafka.record

key.ignore=true
schema.ignore=true

value.converter=org.apache.kafka.connect.json.JsonConverter
value.converter.schemas.enable=false

transforms=StringToJson
transforms.StringToJson.type=com.example.StringToJsonTransformer

elastic.https.ssl.truststore.type=PKCS12
elastic.security.protocol=SSL
elastic.https.ssl.truststore.location=http.p12
```

Kafka Connector에서 Elasticsearch로 데이터를 전송하는 과정에서 JSON데이터가 String타입으로 바뀌는 문제가 있어서 명시적으로 타입을 변환해주는 Plugin(StringToJson.jar)을 제작

# Data Visualization

https://lignah.me/malware_trend/

1. Word Cloud : 주요 악성코드 시그니처 빈도
2. Scatter Plot : 파일타입별 다운로드 수 비교
3. Heatmap : 태그별 악성코드 분포
4. Pie Chart : 파일타입 비율 시각화
5. Geo Map : 국가별 악성코드 분포

# Epilogue

- 이번 프로젝트는 단순 데이터 수집을 넘어, 실시간 보안 데이터 파이프라인을 직접 설계하고 구현한 과정이었음

- 아래와 같은 여러 문제들을 겪고 해결함

- Logstash가 MongoDB를 지원하지 않았기 때문에 MongoDB를 선택
  (지금은 Kafka Connect로 문제를 해결했지만, 현재 지원이 중단된 Logstash for MongoDB 오픈소스를 되살려 보는 과정을 계획하고 있음)

- MongoDB Source Connector의 JSON 구조문제로 Elasticsearch Sink Connector 에서 기대하는 구조와 달라 직접 커스텀 변환 플러그인 제작

- HTTP -> HTTPS 전환 성공함. 작은 시도였지만 서비스 보안을 고려한 행동이었음

- Repository : https://github.com/lignah/malware_trend