산업용 대용량 데이터 처리를 위한 분산 로그 처리 장치 및 방법

송현석, 윤창범, 박택근*

한전KDN 전력ICT연구원

{hyunseok.song.17, changbe0m.yun, *reply_1997}@kdn.com

Distributed Log Processing Device and Method for Industrial Large-Scale Data Processing

Hyun-Seok Song, Chang-Beom Yun, Taek-Keun Park

KDN Electric Power ICT Research Institute

요 약

본 논문은 스마트 팩토리의 확산으로 산업 현장(OT) 내 데이터가 폭증함에 따라, 기존 중앙 집중형 처리 방식은 네트워크 대역폭 포화와 실시간성 저하라는 한계에 직면했다.[1] 본 논문은 이를 해결하기 위해 엣지(Edge) 단에서 로그 수집, 딥 패킷 분석(DPI), 그리고 동적 부하 분산을 수행하는 '산업용 분산 로그 처리 시스템'을 제안한다. Rust 언어 기반의 고성능 엔진을 적용하여 메모리 안전성을 확보하고, 실험을 통해 기존 시스템 대비 3배 이상의 처리량과 58%의 메모리 절감 효과를 입증하였다.

Ⅰ. 서 론

4차 산업혁명과 함께 제조 현장은 OT(Operational Technology)와 IT(Information Technology)가 융합된 초연결 환경으로 진화하고 있다. 수천 개의 IoT 센서와 PLC(Programmable Logic Controller)가 실시간으로 데이터를 쏟아내면서, 이를 안정적으로 수집하고 분석하는 것이 생산 효율성과 보안의 핵심 과제가 되었다.

그러나 현재 주류를 이루는 '클라우드 중심(Cloud-Centric)' 또는 '중앙 집중형' 로그 처리 아키텍처는 다음과 같은 구조적 문제점을 안고 있다. 첫째, 대역폭 비용 및 지연(Latency)은 모든 원시(Raw) 데이터를 중앙 서버로 전송하는 과정에서 막대한 네트워크 트래픽이 발생하며, 이는 실시간 이상 탐지를 저해하는 지연 시간을 유발한다. 둘째, 단일 실패 지점(SPOF) 및 가용성은 중앙 서버나 네트워크 링크에 장애가 발생할 경우, 현장의 데이터가 소실되거나 가시성(Observability)을 잃게 된다. 셋째, 보안 사각지대는 단순한 로그 수집만으로는 Modbus, OPC UA 등 산업용 프로토콜을 이용한 정교한 제어 명령 변조 공격을 탐지하기 어렵다.

본 연구는 이러한 문제를 해결하기 위해 '산업용 분산 로그 처리 장치 및 방법'을 제안한다. 제안하는 시스템은 데이터 발생원과 인접한 엣지 게이트웨이에서 1차적인 로그 정제와 보안 분석(DPI)을 수행하고, 클러스터링을 통해 부하를 분산시키는 구조를 갖는다. 이를 통해 네트워크 비용을 절감하고, 실시간성을 확보하며, OT 보안을 강화하는 것을 목적으로 한다.

Ⅱ. 본론

2.1 관련 연구 (Related Work)

엣지 컴퓨팅과 로그 수집 기술 기존에는 Logstash나 Fluentd와 같은 JVM 또는 Ruby 기반의 수집기가 널리 사용되었으나, 이는 높은 메모리 사용량과 GC(Garbage Collection)로 인한 성능 저하 문제로 인해 저사양 산업용 엣지 장비에 탑재하기 부적합했다. 최근 C 기반의 Fluent Bit나 Rust 기반의 Vector와 같은 경량 수집기가 등장하였으나, Fluent Bit는 메모리 안전성 문제와 복잡한 데이터 변환 로직 구현의 한계가 지적되고 있다.

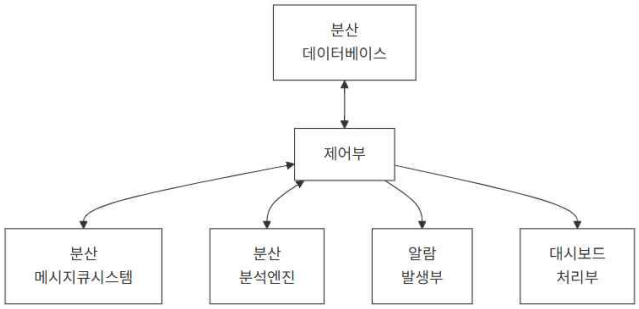
특성	Logstash	Fluentd	Fluent Bit	Vector
개발 언어	Java	Ruby	C	Rust
메모리 사용량	높음 (>1GB)	중간 (~150MB)	매우 낮음 (<50MB)	낮음 (~80MB)
처리 성능	중간	중간	높음	매우 높음
안전성	높음	중간	낮음	매우 높음
데이터 변환	강력함	강력함	제한적	강력함

[표 1] 주요 로그 수집기 기술 비교 분석

산업용 DPI (Deep Packet Inspection) OT 보안을 위해서는 로그뿐만 아니라 네트워크 트래픽의 페이로드를 분석해야 한다. 기존 방화벽은 IP/Port 기반의 차단만 수행하여, 정상 포트(502 등)를 통한 비정상 제어 명령(PLC Stop 등)을 걸러내지 못한다. 따라서 엣지 단에서 경량화된 DPI 기술을 적용하여 실시간으로 패킷을 분석하는 연구가 필수적이다.[2]

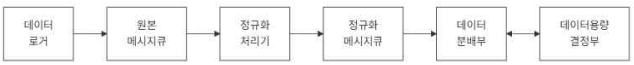
2.2 제안 시스템 설계 (Proposed System Design)

본 연구에서 제안하는 시스템은 엣지 게이트웨이(Edge Gateway)에 탑재되는 소프트웨어 엔진으로, 다음과 같은 핵심 모듈로 구성된다.



[그림 1] 전체 시스템 아키텍처

Rust 기반 고성능 수집 및 처리 엔진 메모리 안전성(Memory Safety)과 고성능을 보장하는 Rust 언어로 개발된 Vector 엔진을 커스터마이징하여 사용한다. 이는 기존 Fluentd 대비 적은 리소스로 대량의 로그를 처리할 수 있으며, VRL(Vector Remap Language)을 통해 엣지 단에서 복잡한 필터링 및 익명화 처리를 수행한다.



[그림 2] 데이터 처리 흐름도

OT 특화 경량 DPI 모듈 네트워크 인터페이스를 미러링하여 Modbus TCP, OPC UA 패킷을 실시간으로 캡처한다. Rust의 nom 파서 라이브러리를 활용하여 제로 카피(Zero-copy) 방식으로 패킷 헤더와 페이로드를 분석, 'Write Coil'이나 'Firmware Update'와 같은 민감한 제어 명령을 식별하여 보안 로그로 변환한다.

동적 부하 분산(Dynamic Load Balancing) 클러스터링 다수의 엣지 게이트웨이가 서로의 상태(CPU, 메모리, 큐 길이)를 가십 프로토콜(Gossip Protocol)로 공유한다. 특정 노드에 트래픽이 집중될 경우, '자원 인지형 최소 연결(Resource-Aware Weighted Least-Connection)' 알고리즘에 따라 인접한 유휴 노드로 로그 처리를 위임(Offloading)하여 시스템 전체의 가용성을 유지한다.[3]

2.3 구현 및 성능 평가 (Implementation & Evaluation)

제안 시스템의 유효성을 검증하기 위해 Raspberry Pi 4 (4GB RAM) 4대로 엣지 클러스터를 구성하고, 초당 10,000건(10k EPS)의 로그와 패킷 트래픽을 생성하여 실험을 수행하였다.

리소스 효율성 비교 기존 널리 사용되는 Fluentd(Ruby)와 제안하는 Vector(Rust) 기반 시스템의 리소스 사용량을 비교하였다.

메모리 사용량은 평균 180MB를 점유한 반면, 제안 시스템은 75MB로 약 58% 절감되었다.

CPU 안정성은 GC 발생 시 순간적인 CPU 스파이크가 관측되었으나, 제안 시스템은 Rust의 효율적인 메모리 관리 덕분에 일정한 CPU 점유율(약 18%)을 유지하였다.

구분	Fluentd (Ruby)	Fluent Bit (C)	Vector
CPU 사용	45%	12%	18%
메모리 사용량	180 MB	35 MB	75 MB
메모리 안정성	불안정	보통	매우 안정

[표 2] 로그 수집기별 리소스 사용량 비교

처리량(Throughput) 및 지연 시간 동일 하드웨어에서 최대 처리 가능한 이벤트 수(EPS)를 측정한 결과, 제안 시스템은 Fluentd 대비 약 3배 높은 처리량을 기록했다. 또한, DPI 기능을 활성화한 상태에서도 패킷당 처리 지연 시간이 평균 0.5ms 미만으로 측정되어, 실시간 제어 망에 영향을 주지 않음을 확인했다.

부하 분산 효과 특정 노드에 과부하(CPU 90% 이상)를 유발했을 때, 제안된 동적 부하 분산 알고리즘이 작동하여 약 40%의 트래픽이 자동으로 인접 노드로 분산되었다. 결과적으로 단일 노드 장애 시 발생하던 데이터 유실(Drop) 현상이 완전히 제거되었다.

Ⅲ. 결론

본 연구에서는 스마트 팩토리 환경의 대용량 데이터 처리를 위해 Rust 기반의 산업용 분산 로그 처리 장치를 제안하였다. 제안 시스템은 엣지 단에서 로그와 네트워크 트래픽을 융합 분석하고, 클러스터링을 통해 부하를 분산함으로써 기존 중앙 집중형 방식의 한계를 극복하였다. 실험 결과, 리소스 사용량 절감과 처리 성능 향상, 그리고 높은 가용성을 입증하였다. 본 기술은 향후 AI 기반의 엣지 이상 탐지 모델과 결합하여, 산업 현장의 지능형 보안 관제 시스템으로 확장될 수 있을 것이다.[4]

참 고 문 헌

[1] E. E. Abel et al., "IoT data analytic algorithms on edge-cloud infrastructure: A review," Digital Communications and Networks, 2023.

[2] B. T. DPI Author, "Privacy-Preserving Traceable Encrypted Traffic Inspection," IEEE Internet of Things Journal, 2024.

[3] M. Merah et al., "Dynamic load balancing of traffic in the IoT edge computing environment," Cluster Computing, 2024.

[4] H. Nizam et al., "Real-Time Deep Anomaly Detection Framework for Multivariate Time-Series Data in Industrial IoT," IEEE Sensors Journal, 2022.