대규모 분산 시스템 환경에서 효율적 로그 관리 시스템에 관한 연구

이상준
숭실대학교

sjlee4628@iteasy.co.kr

A Study on the Efficient Log Management System In Large Distributed System Environment

Lee Sang Joon
Soongsil Univ.

요 약

현대의 IT 시스템은 고도화됨에 따라 대규모 분산 환경에서 다양한 형태의 로그 데이터를 생성하고 있습니다. 일부 어플리케이션 로그는 수집 및 모니터링 시스템을 통해 관리되고 있으나, 여전히 많은 경우 로그 데이터는 분산되어 관리되거나 제대로 관리되지 않는 경우가 많습니다. 특히, 서버에서 생성되는 시스템 로그는 개별적으로 관리되거나 목적별로 상이한 로그 관리 시스템을 사용함으로써 유지보수 및 운영에 어려움을 겪고 있습니다.

본 연구에서는 이러한 문제를 해결하기 위해 어플리케이션 로그와 시스템 로그를 통합적으로 관리할 수 있는 효율적인 로그 관리 시스템을 제안합니다. 제안된 시스템은 대규모 분산 환경에서 로그 데이터 수집, 저장, 처리 및 분석을 통합적으로 지원하며, 로그 관리 효율성을 높이고 유지보수 비용을 절감하는 데 초점을 맞추고 있습니다.

Ⅰ. 서 론

현대의 IT 시스템은 대규모 분산 환경으로 발전하면서 다양한 어플리케이션과 서버 인프라에서 방대한 로그 데이터가 생성되고 있습니다. 이러한 로그 데이터는 시스템 상태 모니터링, 문제 진단, 성능 최적화와 같은 필수적인 운영 활동에 있어 중요한 역할을 합니다. 그러나 로그 데이터의 종류와 수집 방식이 다양하고, 개별적인 관리 시스템이 병존함에 따라 효율적인 로그 관리를 위한 통합된 접근 방식의 필요성이 대두되고 있습니다.

특히, 어플리케이션 로그는 이미 많은 환경에서 중앙 집중식으로 수집 및 관리되고 있으나, 서버에서 생성되는 시스템 로그는 여전히 분산된 방식으로 관리되거나 적절한 관리 체계가 부재한 경우가 많습니다. 이로 인해 로그 데이터의 통합 분석 및 활용이 어려워지고, 로그 관리와 유지보수의 복잡성이 증가하고 있습니다. 이러한 문제는 특히 대규모 분산 시스템 환경에서 심화되며, 로그 데이터를 신속히 처리하고 분석하는 데 있어 운영 효율성을 저하시키는 요인이 되고 있습니다.

본 연구에서는 이러한 문제를 해결하기 위해 각 서버에 에이전트를 설치하여 로그를 중앙 서버로 전달하고, 이를 분석 및 저장하여 통합 관리할 수 있는 효율적인 로그 관리 시스템을 제안합니다. 제안된 시스템은 다음과 같은 구조로 설계되었습니다.

1. 로그 수집 단계: 각 서버에 설치된 에이전트가 로그 데이터를 수집하고, 이를 Kafka 큐에 전달합니다.

2. 로그 처리 단계: Kafka 에 삽입된 로그 데이터를 Kubernetes 환경에서 실행되는 Logstash 가 가져와 분석 및 변환 작업을 수행합니다.

로그 저장 단계: 분석된 로그 데이터를 Elasticsearch 에 저장하여 빠른 검색과 효율적인 관리가 가능하도록 합니다.

이 시스템은 ELK 스택(Elasticsearch, Logstash)을 기반으로 설계되어 높은 확장성과 유연성을 제공하며, 대규모 분산 환경에서도 효율적으로 로그를 수집, 분석, 저장할 수 있을 것으로 기대됩니다. 이를 통해 기존의 분산된 로그 관리 방식에서 발생하는 비효율성을 해소하고, 운영 및 유지보수의 복잡성을 줄일 수 있습니다.

본 연구는 이러한 통합 로그 관리 시스템의 설계와 구현 방안을 제시하고, 실험적 검증을 통해 시스템의 성능과 효율성을 평가함으로써 대규모 분산 시스템 환경에서의 로그 관리 문제를 해결할 수 있는 대안을 제공합니다.

Ⅱ. 통합 로그 관리 시스템

1. 연구 배경 및 필요성

기존 로그 관리 방식의 한계

대규모 분산 환경에서 로그 데이터는 시스템 운영 및 유지보수에 핵심적인 요소입니다. 그러나 기존의 로그 관리 방식은 다음과 같은 한계를 보입니다:

분산된 관리 체계: 서버 로그와 애플리케이션 로그가 독립적으로 관리되어 데이터 통합 및 분석이 어려움.

확장성 문제: 데이터 증가에 따른 시스템 처리 한계.

복잡한 유지보수: 서로 다른 도구와 프로세스를 사용하는 경우 유지보수 부담 증가.

통합 로그 관리의 필요성

통합 로그 관리는 다음의 이점을 제공합니다:

로그 데이터의 중앙 집중화로 통합 분석 가능.

효율적인 리소스 활용 및 시스템 확장성 보장.

관리 및 유지보수의 단순화.

2. 시스템 설계

아키텍처 개요

제안하는 통합 로그 관리 시스템은 다음과 같은 아키텍처로 구성됩니다:

1. 에이전트: 각 서버에 설치되어 로그 데이터를 실시간으로 수집.
2. Kafka: 수집된 데이터를 큐잉하고 실시간으로 처리.
3. Logstash: Kafka 에서 데이터를 가져와 필터링 및 변환.
4. Elasticsearch: 데이터를 저장하고 빠른 검색 및 관리 가능.

주요 구성 요소의 역할

에이전트: 서버에서 로그 데이터를 수집하고 Kafka 로 전송.

Kafka: 로그 데이터를 안정적으로 큐잉하여 고속 처리.

Logstash: 데이터를 필터링하고 Elasticsearch 로 전달.

Elasticsearch: 저장된 데이터를 빠르게 검색하고 관리.

3. 구현 방법

에이전트 설정

각 서버에 에이전트를 설치하여 다양한 로그 데이터를 수집.

JSON 형식으로 데이터를 표준화하고 Kafka 로 전송.

Kafka 설정

로그 유형별 토픽 구성.

데이터 안정성을 위해 리플리케이션 설정.

Logstash 파이프라인

Kafka 로부터 데이터를 가져와 필요한 필터링 및 변환 수행.

Elasticsearch 로 데이터를 전송하여 인덱싱 및 저장.

Elasticsearch 설정

고성능 검색을 위한 인덱스 매핑 및 샤드 구성.

복제본 설정으로 데이터 가용성 확보.

4. 성능 평가

실험 환경

하드웨어: CPU, 메모리, 디스크 I/O 등 명시.

소프트웨어: Kafka, Logstash, Elasticsearch 의 버전 및 설정.

클러스터 구성: 노드 수 및 리소스 분배.

평가 지표

TPS (초당 처리량): 초당 처리 가능한 로그 데이터 수.

처리 시간: 로그 데이터 수집부터 저장까지의 소요 시간.

데이터 손실률: 시스템 장애 상황에서의 데이터 손실 여부.

실험 결과

TPS 비교: 기존 시스템 대비 제안 시스템의 처리량 증가.

처리 시간 감소: 실시간 로그 처리를 통한 지연 시간 단축.

데이터 안정성: Kafka 의 리플리케이션을 통해 데이터 손실 방지.

Ⅲ. 결론

본 연구는 대규모 분산 환경에서 발생하는 다양한 로그 데이터를 효율적으로 관리하기 위한 통합 로그 관리 시스템을 제안하고 이를 구현하였습니다. 기존 로그 관리 방식에서 나타나는 분산 관리, 확장성 부족, 복잡한 유지보수 문제를 해결하기 위해, 에이전트 기반 로그 수집, Kafka 를 활용한 안정적인 메시지 대기열, Logstash 의 데이터 변환, Elasticsearch 를 통한 효율적인 저장 및 검색을 결합한 통합 아키텍처를 설계하였습니다.

실험 결과, 제안 시스템은 기존 방식 대비 초당 처리량(TPS)과 처리 시간, 데이터 손실률에서 우수한 성능을 보였으며, 확장성과 안정성 측면에서도 높은 수준을 입증하였습니다. 이러한 결과는 대규모 분산 환경에서 제안된 통합 로그 관리 시스템이 실질적으로 효과적이고 실행 가능하다는 것을 보여줍니다.

향후 연구에서는 다음과 같은 방향성을 고려할 수 있습니다:

1. 보안 강화: 데이터 암호화, 접근 제어와 같은 보안 기능 통합.

2. 다양한 로그 포맷 지원: 시스템 로그, 네트워크 로그, 애플리케이션 로그 등 다양한 데이터 형식에 대한 지원 확대.

3. 실시간 이상 탐지: 로그 데이터를 활용한 머신 러닝 기반의 실시간 이상 탐지 기능 추가.

4. 운영 효율성 향상: 자동화된 설정 및 관리 도구 개발.

본 연구는 대규모 분산 시스템에서의 로그 관리 효율성을 향상시키는 데 기여하며, 향후 관련 연구 및 산업적 적용 가능성을 높이는 데 중요한 기초 자료가 될 것입니다.

ACKNOWLEDGMENT

Put sponsor acknowledgments.

참 고 문 헌

[1] NGUYEN, Van Nam; TRAN, Van Cuong. An efficient log management system. VNU Journal of Computer Science and Communication Engineering, 2016, 32.2: 43-48.

[2] L. Chen, J. Liu, M. Xian and H. Wang, "Docker Container Log Collection and Analysis System Based on ELK," 2020 International Conference on Computer Information and Big Data Applications (CIBDA), Guiyang, China, 2020, pp. 317-320.

[3] Y. Zong, "Distributed log collection for business processes based on ELK architecture," 2023 IEEE 2nd International Conference on Electrical Engineering, Big Data and Algorithms (EEBDA), Changchun, China, 2023, pp. 1523-1529.

[4] J. Hyun and H. Kim, "Security Operation Implementation through Big Data Analysis by Using Open Source ELK Stack," Journal of Digital Contents Society, vol. 19, no. 1, pp. 181-191, Jan. 2018.