Apache Lucene, ELK stack and Telcom applications

CS410 – Text Information Systems Fall 2020

Technology Review

Suraj Bisht

surajb2@illinois.edu

Introduction

During recent years Elasticsearch analytics engine is being used by many commercial solutions. This tech paper is focused on ELK stack offered by elastic and specific use cases in Telecom domain. System and application logs offer unique insight into telecom operation and system health. Telecom traffic pattern varies a lot hence operation efficiency, anomaly and automation can be achieved using analyzing recent and historical traffic pattern. Elastic solution which includes beats, Logstash, Elasticsearch and Kibana offers end-to-end capability from system data collection, filter & data storage, search, visualization to partial machine learning capability for anomaly detection.

Background

Lucene is a full-text search library in Java which makes it easy to add search functionality to an application or website. It does so by adding content to a full-text index. It then allows you to perform queries on this index, returning results ranked by either the relevance to the query or sorted by an arbitrary field such as a document's last modified date.

Lucene is able to achieve fast search responses because, instead of searching the text directly, it searches an index instead. This would be the equivalent of retrieving pages in a book related to a keyword by searching the index at the back of a book, as opposed to searching the words in each page of the book. This type of index is called an inverted index, because it inverts a page-centric data structure (page->words) to a keyword-centric data structure (word->pages).

In Lucene, a Document is the unit of search and index. An index consists of one or more Documents. Indexing involves adding Documents to an IndexWriter and searching involves retrieving Documents from an index via an IndexSearcher. A Lucene Document doesn't necessarily have to be a document in the common English usage of the word. For example, if you're creating a Lucene index of a database table of users, then each user would be represented in the index as a Lucene Document. Apache Lucene is a free and open source search engine software library, originally written in Java language.

Elasticsearch is an open source, full-text search and analysis engine, based on the Apache Lucene search engine. Logstash is a log aggregator that collects data from various input sources, executes different transformations and enhancements and then ships the data to various

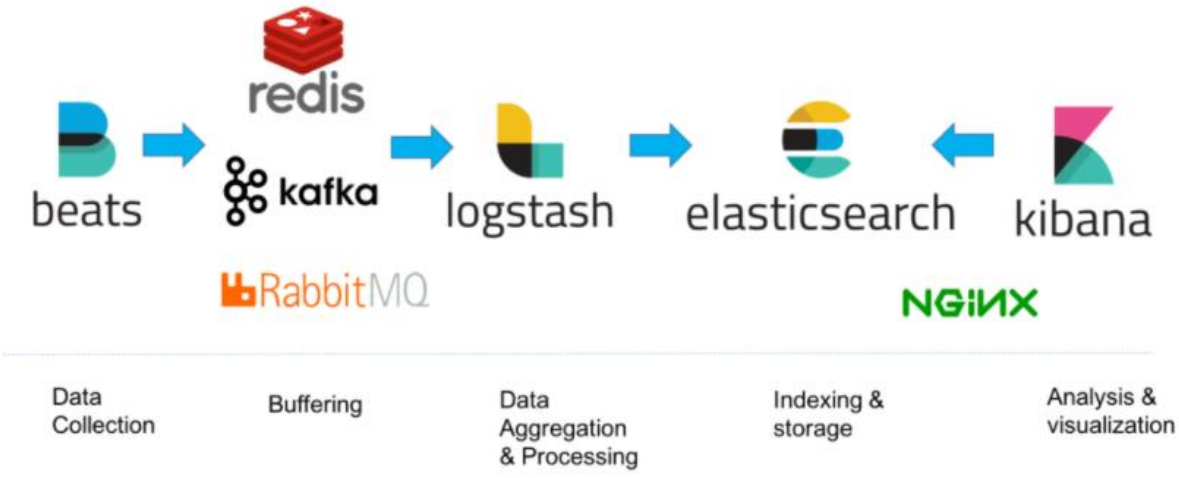
supported output destinations. Elasticsearch offer shard-based implementation which helps in scaling the solution. Kibana is a visualization layer that works on top of Elasticsearch, providing users with the ability to analyze and visualize the data. And last but not least — Beats are lightweight agents that are installed on edge hosts to collect different types of data for forwarding into the stack.

ELK offers many ready to use beats, Metricbeat, Auditbeat, Packetbeat, filebeat etc for different types of log collection and corresponding visualizations that helps quickly analyzing patterns and insights

Logstash offers memory or storage-based pipelines which support buffering to prevent data loss for scenarios when output towards Elasticsearch or other destination is down. Logstash pipeline configuration offers filter that can be used to update output format and create different index pattern depending upon incoming source type, beat or application. Pipeline configuration also supports forking data towards multiple destination based on various

Implementation

Production ready complex solution with high availability, resilience and reliability supporting critical telecom solution includes following components



Multiple solutions can be offered using ELK stack

- Telecom core solutions includes hundreds of separate applications, but applications are linked together, and those specific patterns can only be analyzed and visualize using log and data aggregation.
- For a virtualized solution corelating error pattern from platform and application running on these virtualized or Kubernetes platform especially private cloud.
- Closed-loop automation

Most of the telecom traffic are flows through multiple services and applications hence identifying anomaly for one application can potentially help in analyzing impact on other services. Various techniques and algorithms can be used to identify root cause and implementing system recovery without manual intervention.

Telecom solution are mostly hosted on openstack or vmware based private cloud. Elasticsearch based solutions are being used to corelated platform and application logs that offer unique insight on overall system health. Future can be predicted based on historical log and stats pattern during some hardware failure. Action like live migration from faulty hardware can be provide auto-recovery

Application current load pattern plus historical metrics and logs are used to detect anomaly and alert/alarm can be raised to automate or prevent potential service outage.

Elasticsearch support search within same cluster as well cross-cluster which is perfect for solutions deployed across multiple solution. Independent Elasticsearch cluster can be deployed for collecting logs/data from locally deployed applications and avoid costly replication across sites. At the same time, it supports on-demand cross-cluster aggregation for searching and combination results from applications across all the sites.

Closed loop automation, which is generally achieved by data collection, analyzing & correlation, problem detection, root cause analysis and correction. ELK solution can be used for Closed loop automation for telecom as well as any other domain solution.

Despite offering complete solution, ELK stack have some limitation w.r.t dimensioning and capacity planning for handling large amount of data especially more than trillion events or petabyte data size. For solutions which involved decision making based on historical data calculation Elasticsearch are some limitation to process such large data in near real-time may not always be feasible.

Conclusion

Elasticsearch strength in log aggregation and efficient search within same cluster or cross cluster makes solution best for analyzing system behavior in almost real-time and provide opportunity to develop closed-loop automation for small to medium size application(s).

Reference

https://www.elastic.co/

https://en.wikipedia.org/wiki/Apache_Lucene

https://en.wikipedia.org/wiki/Elasticsearch