Poster: A Real-time Dataflow Process Platform for Network Forensics

Fei Xu[1,2], Gang Xiong[1], Zhen li[1], Junzheng Shi[1]
[1]Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
[2]Law School, Renmin University of China, Beijing, China
{xufei, xionggang, lizhen, shijunzheng }@iie.ac.cn

***Abstract*—Today's network forensics are struggling to bring together enormous volumes of heterogeneous data from all kinds of logs and network traffic, and to analyze the huge amount of data in real time. In this paper, we present the design, implementation, and evaluation of a real time dataflow process platform using a combination of open source tools that captures, retains and analyze high amount of network activities and logs to further support network forensic. Our proposed platform is now being used in a large company in China and has been proved to achieve high performance and can be used in real time with the maximum throughput of around 5Gb/s.**

Keywords—Network Forensics; Dataflow Process Platform

I. INTRODUCTION

Todays network equipment generates high-quality logs of tens of thousands of events per second. Network forensic system needs to efficiently integrate the disparate sources of data (e.g., NIDS, firewalls, NetFlow data, service logs, packet traces) that investigations often involve. The problems they need to deal with includes: 1) Huge data amount, which requires high input and output ability. 2) Enormous data types, which requires data normalization process. 3) Real-time interactive data analysis and 4) Data presentation.

In order to solve those problems, many studies have been done. Most famous one is VAST [1], A Unified Platform for Interactive Network Forensics, proposed by Matthias Vallentin. There main focus is to provide both continuous ingestion of voluminous event streams and interactive query performance, while our focus is to provide companies and organizations fast and easy way to deploy real time dataflow process platform to better receive, normalize, store, and analyze network data. Our proposed platform uses a combination of open source big data analysis tools, achieve better performance while solve real time data process problems.

II. PLATFORM ARCHITECTURE

A. Modules

We used a combination of different open source tools to set up our platform. Main modules, their functions, and the relationship between different modules are shown by Figure 1. Detailed description is as follows:

1) Data Bus: Kafka is used as data bus. Kafka is a distributed, partitioned, replicated commit log service. It provides the functionality of a messaging system, but with a unique design. A single Kafka broker can handle hundreds of megabytes of reads and writes per second from thousands of clients. Kafka is designed to allow a single cluster to serve as the central data backbone for a large organization. It can be elastically and transparently expanded without downtime.

2) Extract-Transform-Load: Most data are collected from different source systems and each separate system may use a different data organization and/or format. Elasticsearch, Logstash and Kibana will be responsible for data extract and presentation. Data normalization needs to be done before data being stored in databases. Data normalization contains data extract, data transform and data loads, also known as ETL. ETL consumes data from the Kafka queue and extract, transform and load data. We implemented Logstash to perform the ETL.

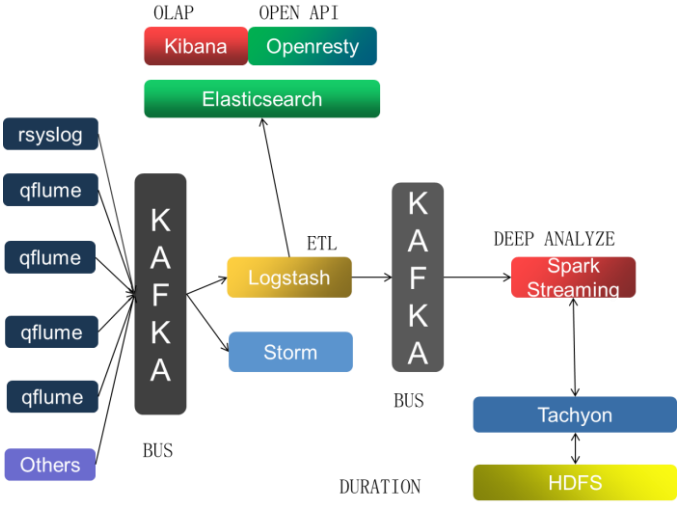

Fig. 1. Platform Architecture

3) Data Duration: There has been an increasing demand of centralized systems to store all the information so that users can retrieve the information as necessary. For post hoc forensic analysis, raw data needs to be stored for a period of time for afterward correlating analysis. Intermediate and final analysis results also need to be stored. Data storage provides a platform to analyze data at different levels of specifications.we choose HDFS as data warehouse to store collected data for data duration purpose and further deep analysis.

4) Deep Analyze: Spark Streaming is perfect for data deep analyze, is a fast and general engine for large-scale data processing. And Spark Streaming in our system is built on Tachyon and will interact with HDFS.

B. Other Frameworks

We also used a distributed architecture Apache Mesos and build other applications on top of it, including Marathon, Chronos, Spark, Tachyon, HDFS, etc, and other containers, to support different system requirements. The frameworks are shown in Figure 2.

Mesos is a distributed systems kernel which abstracts CPU, memory, storage, and other compute resources away from machines, enabling fault-tolerant and elastic distributed systems to easily be built and run effectively.

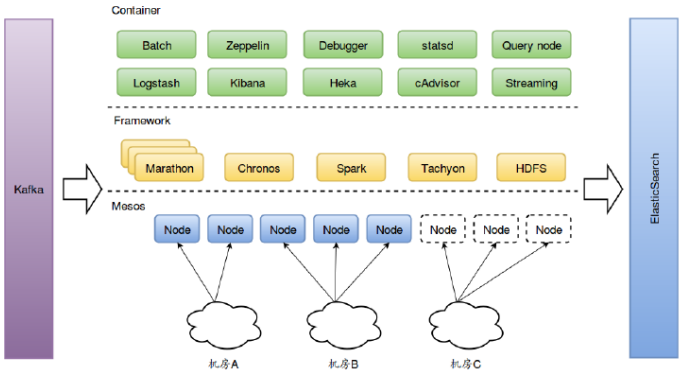

Fig. 2. Other Frameworks

Tachyon is a memory-centric distributed storage system enabling reliable data sharing at memory-speed across cluster frameworks. Marathon is a production-grade container orchestration platform for Apache Mesos.

III. IMPLEMENTAION

We have been using the proposed architecture in the department of operation and maintenance in a large company in Beijing, China for several months for forensic analysis. The analyzer can analyze all the log data and network flow data in real time, and also the data is stored in HDFS for further forensic analysis use.

Fig. 3. Traffic being processed one day

For the performance, as shown in Figure 3, the maximum traffic being processed one day is 5.182 Gb/s, and average traffic being processed is 3.258Gb/s in this day. Also, the performance of the proposed architecture is very stable and elastic, scalable.

Figure 4 shows in real time that how many instances are runing and the data flow in the system, data flow from Kafka to Logstash to ElasticSearch to Kafka. And if any part of the system is having problem, it will be shown on this digram in real time. In Figure 4, there are 31 logstash instances running and 1 Kibana instances running.

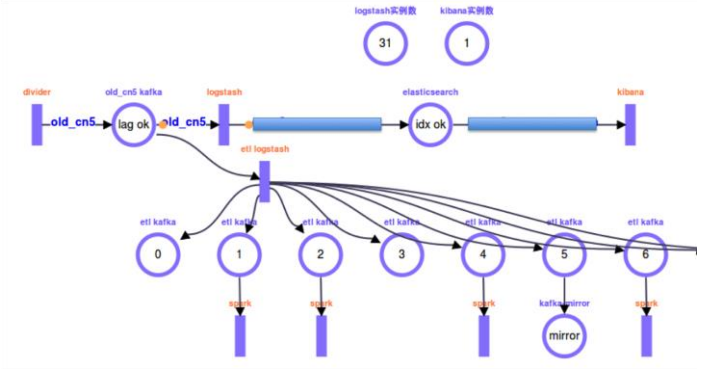

Fig. 4. Real- time system data flow

IV. CONCLUSION AND FUTURE WORK

In this paper, we proposed a unique architecture for real time network forensics in big data environments by using a combination of open source tools. The proposed architecture has been used by a large company in China and has been proven to be good in performance and stability as well as flexibility. Many other containers and applications can be easily added onto this architecture. Meanwhile, this architecture is easy to deploy and maintain for companies and organizations.

Our future work include develop performance optimization manage tools, to further meet the requirement of network forensic requirement. We will also add composing and indexing modules onto our system to support the payload analyze requirement of network forensics while saving the storage spaces.

ACKNOWLEDGMENT

This work is supported by the Strategic Priority Research Program of the Chinese Academy of Sciences (XDA06030200) and Beijing Natural Science Foundation（4164089）.

[1] M. Vallentin, V. Paxson, and R. Sommer. “VAST: A Unified Platform for Interactive Network Forensics” 13th USENIX Symposium on Networked Systems Design and Implementation (NSDI ’16). March 16–18, 2016 • Santa Clara, CA, USA

[2] P. Giura and N. Memon. “NetStore: An Efficient Storage Infrastructure for Network Forensics and Monitoring” Recent Advances in Intrusion Detection. Springer Berlin Heidelberg, 2010: 277-296.

[3] https://www.elastic.co/products/elasticsearch/

[4] http://www.tachyonproject.org

[5] https://www.elastic.co/webinars/introduction-elk-stack

[6] http://kafka.apache.org

[7] https://mesosphere.github.io/marathon/

[8] http://spark.apache.org