Implementing a Streamed Distributed Data Processing Pipeline

Degree programme: BSc in Computer Science | Specialisation: Data Engineering
Thesis advisor: Dr. Souhir Ben Souissi
Expert: Prof. Dr. Torsten Braun

In this bachelor‘s thesis, a pipeline was created with the goal to aggregate a continuous stream of mobile roaming packets. The pipeline parses the binary packets, matches them into full transactions, detects time-outs and aggregates the results in multiple KPIs.

Context

When people or devices use a mobile network they‘re not subscribed to, they are roaming. Comfone AG is a roaming provider for mobile network operators. Comfone offers services to these operators to operate, monitor and improve their roaming services for their customers. One of these services is, providing insight into the roamers‘ successful, erroneous and timed-out connections. To do this roaming packets are analyzed and aggregated continuously. Comfone observes a continuous rise in traffic volume. In order to cope with the increased traffic, Comfone is looking into distributed systems.

Goal

The goal of the project is to create a pipeline that consumes the packets and creates the aggregations on a distributed system. The main focus lies in being able to distribute the load across multiple nodes without manual changes to the code.

The Pipeline

As seen in the picture, the pipeline is divided into multiple tasks, each with its own distinct purpose and responsibilities. The data arrives chunked into larger files, which are split into individual messages. The messages are then parsed into more readable objects. Depending on the protocol contained in the packet different classes are produced. The packets are correlated into transactions which form the basis of the statistics. A transaction can either be successful, erroneous or timed out. Detecting the time-outs relies on having timers that fire when no answer was received in a given time window. Once transactions are created, they are enriched with additional information. These final transactions are aggregated into fixed length time windows of 5 minutes, 1 hour and 24 hours. During the thesis, only the 5 minutes aggregation was produced. The final statistics are stored in a MongoDB cluster and are immediately available to the customers in Comfones customer portal.

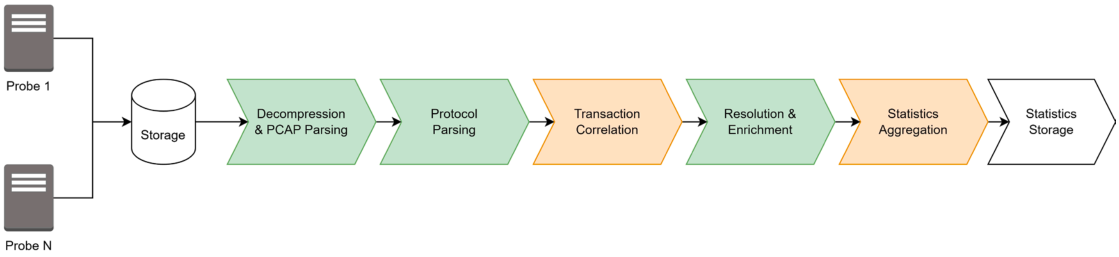
Results

The statistics produced by the new pipeline are very close to the existing solution. In terms of speed, the pipeline is up to 20% faster when run on a single node. Tests were made to see how the overhead of distribution is. The distribution adds roughly a 20% performance loss without adding additional hardware. This means that when doubling the hardware capacity roughly an 80% gain in performance can be expected.

Conclusion

This thesis showed that it‘s possible to distribute the processing of the data quite easily. The distribution will require a complete rewrite in the chosen framework and will encompass a lot of work. However, if the rise in traffic continues as observed over the past years, it will overwhelm a single node eventually and distribution will be mandatory.

Christof Flück



Pipeline Visualisation