Project Proposal: End to End Tracing, Ceph Tracing

Golsana Ghaemi[1], Bowen Song[1], Oindrilla Chatterjee[1], Aditya Singh[1]

[1]Department of Electrical and Computer Engineering
[1]Boston University

1 Vision and Goals Of The Project

This project focuses on enabling a strong and open source tracing infrastructure for Ceph, a novel open source high-performance distributed software storage. The focus of the implementation is to replace Blkin tracing infrastructure with Jaeger for enabling an always-on and open source end to end tracing feature for Ceph.

2 Users/Personas Of The Project

Modern Internet services are often implemented as complex, large-scale distributed systems. These applications are constructed from collections of software modules and could span many thousands of machines across multiple physical facilities. Knowing the system behavior and reasoning about performance issues are invaluable in such environments [1]. Therefore, the obvious use is to allow system engineers and researchers better understand the infrastructure and keep the system at highest efficiency.

3 Scope and Features Of The Project

The project is concerning two tracing tools for tracking requests in Ceph: BlinKin and Jaeger. As a basic tracing infrastructure BlKin is thought to be naive and less effective in terms of always-on to continuously capture traces and on top of that it is simply an addition to Ceph, just applicable to that without any open source community behind it. For a more sophisticated approach, the project is dedicating in introducing Jaeger as a strong and open source tracing infrastructure for enabling end to end tracing and replacing Blkin with it.

4 Solution Concept

Ceph is an open source storage software designed to provide highly scalable object, block and file-based storage under a unified system [2]. As an open source distributed file system, Ceph emphasizes its performance, reliability and scalability with its novel approach towards metadata and improvements to file system principles. A Ceph Storage Cluster requires at least one Ceph Monitor, Ceph Manager, and Ceph OSD (Object Storage Daemon). The Ceph Metadata Server is also required when running Ceph Filesystem(CephFS) clients [3].The novelty in distributing metadata workload is dependent on object-based storage which allows direct communication between storage unit and client.

An evolution in committing Reliable Automatic Distributed Object Storage (RADOS) and its sibling interfaces helps to achieve linear scaling capacity and performance in terms of availability, throughput rate, and fail recovery. Beside Ceph specific interface (RADOS), there are three other standard interfaces that work with RADOS(called as clients): CephFS for handling POSIX, Rados Block Device(RBD) for handling images (and virtual machines), and Radios Gateway (RGW) for handling REST API requests. [4]

The importance of Ceph is in its advancement, its scalability and the volume of acceptance from the industry. Due to its importance, it crucial to know Cephs behavior. Tools such as tracing infrastructures that help us understanding system behavior and reasoning about performance issues are invaluable.

This project wishes to enable end-to-end tracing, from the request issue time till the time that is completed. The default tracing tool for ceph is Ceph Blkin which is considered as a naive one and is going to be replaced by Jaeger. The effort is to better study the anomaly, steady-state problem, distributed profiling, and resource attribute within the system [5].

4.1 Blkin Tracing System

Blkin is a library which follows the tracing semantics of Googles Dapper. It allows us to trace applications using LTTng, an open source tracing framework for Linux [6]. The major drawbacks of Blkin is its offline tracing (i.e. for tracing Ceph using Blkin, it should be started and then stopped and traces are collected for that specific time slot). Of course this way is not appropriate for one system (like Ceph) that is always running. In addition, Blkin is specified for Ceph, there is no specific open source community to work on it independently and improve it. So the solution can be replacing Blkin with other efficient and open source tracing tool like Jaeger.

4.2 Jaeger Tracing System

Jaeger, inspired by Dapper and OpenZipKin, is a distributed tracing system released as open source by Uber Technologies. It can capture traces, process them and finally visualize them and has its own specific interface for doing that. On each node of the cloud, Jaegers agent should be installed and tracing point of the user application should be in Jaegers syntax (format).

The most important pros of Jaeger is that it is supported by an open source community and is improved continuously. It is literally important to have such a tracing tool in a live community like Ceph.

5 Acceptance Criteria

1. A weighted decision has been made to determine if Jaeger can be used to replace the existing Blkin infrastructure.
2. Jaeger has been introduced as a tracing infrastructure in Ceph.
3. We are able to capture traces for Ceph using Jaeger.
4. We can visualize the traces thereby generated.

6 Release Planning

The project would be implemented and delivered incrementally at the end of each iteration

1. Compilation of the project proposal
2. Showing traces using Blkin (making instances in OpenStack environment, Deploying Ceph on instances in OpenStack, Compiling Blkin for Ceph, Running Ceph with Blkin)
3. Understanding the Blkin architecture, replacement of Blkin with Jaeger
4. Presenting the plan and starting replacing Blkin with Jaeger
5. Demonstration of capturing traces using Jaeger, system integration, testing, bug-fixing and user documentation completion

References

[1] B. H. Sigelman, L. A. Barroso, M. Burrows, P. Stephenson, M. Plakal, D. Beaver, S. Jaspan, and C. Shanbhag, "Dapper, a large-scale distributed systems tracing infrastructure," Google, Inc., Tech. Rep., 2010. [Online]. Available: https://research.google.com/archive/papers/dapper-2010-1.pdf

[2] Red Hat, Inc. (2017) Ceph homepage. [Online]. Available: https://ceph.com

[3] ——. (2016) Intro to Ceph Ceph Documentation. [Online]. Available: http://docs.ceph.com/docs/master/start/intro/

[4] Sage A. Weil, et al., "Ceph: a scalable, high-performance distributed file system," *OSDI 06 Proceedings of the 7th symposium on Operating systems design and implementation*, 2006.

[5] Raja R. Sambasivan, et al., "So, you want to trace your distributed system? Key design insights from years of practical experience," *Carnegie Mellon University Parallel Data Lab Technical Report CMU-PDL-14-102*, April, 2014.

[6] Red Hat, Inc. (2016) Tracing Ceph With BlkKin Ceph Documentation. [Online]. Available: http://docs.ceph.com/docs/master/dev/blkin/