Shiyue Hou
Virtualization
Paper Review: Serverless in the Wild: Characterizing and Optimizing the Serverless Workload at a Large Cloud Provider

Summary:
Function-as-a-Service (FaaS) is a type of cloud-computing service that allow user to execute program without extra configuration. With FaaS, user only needs to focus on the code itself. According to code recourse requirements, the cloud service provider will provide the physical hardware, virtual machine operating system and web server software management automatically. The FaaS is a subset of serverless. FaaS is central technology of serverless and focus on event-driven computing model. The serverless most concentrate on service part and services configuration; management are invisible for user.

For the provider, it prefers to find the high-performance function with low resource consume. There are three ways to find the high-performance function, the one is trigger program from the memory, because the program execute from memory is faster than in disk. Another one is main all potential execute program in memory. The last one is function are used by widely resource. But there is no characteristics can be reference at FaaS workload. The paper proposes to characterize the production workload by using characterize the real function, invoked type, invocation frequencies and model.

More than that, this paper proposes a new practical resource management policy for reducing the number of cold start executions by using a small histogram to keep track the recent function inter-invocation time.

Strengths:
1. This paper proposes characterize production workload and researcher can according to characterized workload to generate realistic trace to improve FaaS workload.
2. This paper presents a new resource management policy to improve FaaS workload performance by using reduce the number of cold start execution.
3. The resource management policy implements at simulation and experiment at Apache Open Whisk FaaS real device.
4. Within this paper, author release a bunch of traces at github as open sources, which can benefit for research community.

Weaknesses:
1. As data collection, for the execution time, author only select a certain range of invocations. The wild a range of invocations do not consider.
2. From the Microsoft Azure confidential perspective, some data such as total number of invocation and functions are not absolute.
3. In this paper experiment, for data collection, the data are available only for execution time more than one minute.
4. For the cold start analyze, the cold start from concurrency problems doesn’t be considered.