World Academy of Science, Engineering and Technology
International Journal of Information and Communication Engineering
Vol:12, No:03, 2018

Automatic Aggregation and Embedding of Microservices for Optimized Deployments

Authors : Pablo Chico De Guzman, Cesar Sanchez

Abstract : Microservices are a software development methodology in which applications are built by composing a set of independently deploy-able, small, modular services. Each service runs a unique process and it gets instantiated and deployed in one or more machines (we assume that different microservices are deployed into different machines). Microservices are becoming the de facto standard for developing distributed cloud applications due to their reduced release cycles. In principle, the responsibility of a microservice can be as simple as implementing a single function, which can lead to the following issues: - Resource fragmentation due to the virtual machine boundary. - Poor communication performance between microservices. Two composition techniques can be used to optimize resource fragmentation and communication performance: aggregation and embedding of microservices. Aggregation allows the deployment of a set of microservices on the same machine using a proxy server. Aggregation helps to reduce resource fragmentation, and is particularly useful when the aggregated services have a similar scalability behavior. Embedding deals with communication performance by deploying on the same virtual machine those microservices that require a communication channel (localhost bandwidth is reported to be about 40 times faster than cloud vendor local networks and it offers better reliability). Embedding can also reduce dependencies on load balancer services since the communication takes place on a single virtual machine. For example, assume that microservice A has two instances, a1 and a2, and it communicates with microservice B, which also has two instances, b1 and b2. One embedding can deploy a1 and b1 on machine m1, and a2 and b2 are deployed on a different machine m2. This deployment configuration allows each pair (a1-b1), (a2-b2) to communicate using the localhost interface without the need of a load balancer between microservices A and B. Aggregation and embedding techniques are complex since different microservices might have incompatible runtime dependencies which forbid them from being installed on the same machine. There is also a security concern since the attack surface between microservices can be larger. Luckily, container technology allows to run several processes on the same machine in an isolated manner, solving the incompatibility of running dependencies and the previous security concern, thus greatly simplifying aggregation/embedding implementations by just deploying a microservice container on the same machine as the aggregated/embedded microservice container. Therefore, a wide variety of deployment configurations can be described by combining aggregation and embedding to create an efficient and robust microservice architecture. This paper presents a formal method that receives a declarative definition of a microservice architecture and proposes different optimized deployment configurations by aggregating/embedding microservices. The first prototype is based on i2kit, a deployment tool also submitted to ICWS 2018. The proposed prototype optimizes the following parameters: network/system performance, resource usage, resource costs and failure tolerance.

Keywords : aggregation, deployment, embedding, resource allocation
Conference Title : ICWS 2018 : International Conference on Web Services
Conference Location : Madrid, Spain
Conference Dates : March 26-27, 2018