i2kit: A Tool for Immutable Infrastructure Deployments

Authors : Pablo Chico De Guzman, Cesar Sanchez

Abstract : Microservice architectures are increasingly in distributed cloud applications due to the advantages on the software composition, development speed, release cycle frequency and the business logic time to market. On the other hand, these architectures also introduce some challenges on the testing and release phases of applications. Container technology solves some of these issues by providing reproducible environments, easy of software distribution and isolation of processes. However, there are other issues that remain unsolved in current container technology when dealing with multiple machines, such as networking for multi-host communication, service discovery, load balancing or data persistency (even though some of these challenges are already solved by traditional cloud vendors in a very mature and widespread manner). Container cluster management tools, such as Kubernetes, Mesos or Docker Swarm, attempt to solve these problems by introducing a new control layer where the unit of deployment is the container (or the pod — a set of strongly related containers that must be deployed on the same machine). These tools are complex to configure and manage and they do not follow a pure immutable infrastructure approach since servers are reused between deployments. Indeed, these tools introduce dependencies at execution time for solving networking or service discovery problems. If an error on the control layer occurs, which would affect running applications, specific expertise is required to perform ad-hoc troubleshooting. As a consequence, it is not surprising that container cluster support is becoming a source of revenue for consulting services. This paper presents i2kit, a deployment tool based on the immutable infrastructure pattern, where the virtual machine is the unit of deployment. The input for i2kit is a declarative definition of a set of microservices, where each microservice is defined as a pod of containers. Microservices are built into machine images using linuxkit —- a tool for creating minimal linux distributions specialized in running containers. These machine images are then deployed to one or more virtual machines, which are exposed through a cloud vendor load balancer. Finally, the load balancer endpoint is set into other microservices using an environment variable, providing service discovery. The toolkit i2kit reuses the best ideas from container technology to solve problems like reproducible environments, process isolation, and software distribution, and at the same time relies on mature, proven cloud vendor technology for networking, load balancing and persistency. The result is a more robust system with no learning curve for troubleshooting running applications. We have implemented an open source prototype that transforms i2kit definitions into AWS cloud formation templates, where each microservice AMI (Amazon Machine Image) is created on the fly using linuxkit. Even though container cluster management tools have more flexibility for resource allocation optimization, we defend that adding a new control layer implies more important disadvantages. Resource allocation is greatly improved by using linuxkit, which introduces a very small footprint (around 35MB). Also, the system is more secure since linuxkit installs the minimum set of dependencies to run containers. The toolkit i2kit is currently under development at the IMDEA Software Institute.

Keywords : container, deployment, immutable infrastructure, microservice