Containers in OpenStack

Containers Team Proposal Overview
2014-07-29
Containers in OpenStack

Nova

Containers Service

Scheduler (Gantt)

REST

Map

Compute Host

Instance

Instance

Instance

OSC

C = Container
R = Agent Relay
A = Container Agent
Nova Instances

• Any Nova Instance
  – VM
  – Bare Metal
  – Container

• Run Agent for advanced features
  – Install agent with cloud-init or bake into image
Containers Service

– High Level API
  • Uses concepts of image, cgroup, namespace
  • “docker run” equivalent
    – kill, remove, start, stop

– Map
  • Maintains instance <-> container relations

– Create Nova instances as needed
What to Remember

• Nova API remains the same for users
• Basic containers in Nova with virt drivers
  – nova-docker
  – libvirt/LXC
• Containers Service
  – For container specific functionality
  – Creates nova instances with agents on-demand
  – Transparently nests containers for easy usage
Phase II Plans

- Heat Resource for Containers Service
- Use Gantt to schedule containers on instances
- Glance integration for images
- Emit usage events
Open Questions

• Out of scope for today
  – Encrypted transport to/from agent
    • Model from vnc bp
  – Unprivileged containers (nesting)
Docker Client Use Case

$ source ./openrc
$ export DOCKER_SERVER=https://...
$ docker run -p pub:222:22 -d foo /usr/sbin/sshd -D DEADBEEF
$ osc container-show DEADBEEF | grep ports
ports: [12.34.56.78:222]
$ ssh -p 222 12.34.56.78
foo$
OSC Client Use Case

$ source ./openrc
$ osc container-create --port pub:222:22 --daemon --image <uuid> --cmd "/usr/sbin/sshd -D"
$ osc container-show DEADBEEF | grep ports
ports: [12.34.56.78:222]
$ ssh -p 222 12.34.56.78
foo$
Credits

• Adrian Otto adrian.otto@rackspace.com
• Andrew Melton andrew.melton@rackspace.com
• Chuck Short chuck.short@canonical.com
• Eric Windisch ewindisch@docker.com
• Harris Rick rconradharris@gmail.com
• Kirill Kolyshkin kir@openvz.org
• Thomas Maddox thomas.maddox@rackspace.com