

Title: Docker Swarm CPU overload on deploy with Spring Boot containers

Post Body:

I have created a number of Spring Boot application, which all work like magic in isolation or when started up one of the other manually.

My challenge is that I want to deploy a stack with all the services in a Docker Swarm.

Initially I didn't understand what was going on, as it seemed like all my containers were hanging.

Turns out running a single Spring Boot application spikes up my CPU utilization to max it out for a good couple of seconds (20s+ to start up).

Now the issue is that Docker Swarm is launching 10 of these containers simultaneously and my load average goes above 80 and the system grinds to a halt. The container HEALTHCHECKS starts timing out and eventually Docker restarts them. This is an endless cycle and may or may not stabilize and if it does stabilize it takes a minimum of 30 minutes. So much for micro services vs big fat Java EE applications :(

Is there any way to convince Docker to rollout the containers one by one? I'm sure this will help a lot.

There is a rolling update parameter - <https://docs.docker.com/engine/swarm/swarm-tutorial/rolling-update/> - but it does not seem applicable to startup deployment.

Your help will be greatly appreciated.

I've also tried systemd (which isn't ideal for distributed micro services). It worked slightly better than Docker, but have the same issue when deploying all the applications at once.

Initially I wanted to try Kubernetes, but I've got enough on my plate and if I can get away with Docker Swarm, that would be awesome.

Thanks!

Accepted Answer: None

Highest Rated Answer: None