Groveling before Grouper

From University of Michigan

Before containerizing Grouper I thought I was a fairly-well seasoned identity and access management engineer. (Because I'm a big guy my friends might say I'm well-marbled, but that's another story.)

So when I approached the installation of containerized Grouper I thought I should be able to knock it out in a couple of weeks.

Boy, was I wrong!

I was completely new to containerization. To further complicate matters, containerized Grouper had been created for use with Docker, yet the University of Michigan's platform of choice for containerization is OpenShift.

Working with our local container gurus I had to get into the "container mindset": nothing specific about the environment should be in the container itself. Control everything through environment variables and secrets. I also had to tease apart the differences between Docker and OpenShift.

It was maddening.

It took me a month to develop a process to "bake" into a container the stuff that the Docker compose functionality does automatically.

Once I finally built the images and deployed them to OpenShift successfully, I felt immense pride. However, as pride does, it goeth before the fall.

Disappointingly, containerized Grouper still didn't work. I was under the misapprehension that once I deployed the images to OpenShift, Grouper would magically open up, much like the scene in the movie The Davinci Code when Robert Langdon (Tom Hanks) and Sophie Neveu (Audrey Tautou) enter the code to retrieve the cryptex. Unlike them, I was left with disappointment, frustration, and sadness.

It turns out that simply running kompose convert (which I had stumbled upon, miraculously) and importing all the deployment configurations, routes, and services into OpenShift would not do the trick. I had to get into the nitty-gritty of OpenShift's routing and services architecture myself.

It was a cold January day when I finally configured the routes and services in some meaningful way and was able to retrieve the Grouper service provider's metadata. Progress! And about an hour later, I was finally able to see the Grouper UI, albeit over an unencrypted connection.

To actually log into Grouper successfully, though, would take me another three weeks. I eventually discovered that I had inadvertently shot my own foot, then hit it with a hammer a couple of times: when I had first started working on Grouper, I had modified the services.xml files in an inconsistent and absurd manner.

Once I edited them consistently I was finally able to log into Grouper! Oh joy! Oh bliss!

But never one to rest on my laurels I felt compelled to move forward. Next: implement end-to-end SSL. As it turns out, the solution to SSL was a checkbox and a pull-down menu. To get to the correct combination of clicks, though, took another two weeks.

What's next? In the next two weeks I hope to have containerized Grouper pointing to our development LDAP and MySQL servers.

What have I learned from containerized Grouper so far?

- Despite my advanced old age, I can still learn, albeit it seems a bit more slowly.
- Do not take new technologies for granted.
- Even though the technology may be new, there are probably still parts of it which function similarly to technologies with which I am very familiar.
- Be patient. Chunk what you hope to accomplish into meaningful spoonfuls so as to not get frustrated.

For the interested (or morbidly curious), I am putting together a run book of my travails. It should be available soon.

In conclusion, may all your Grouper pods have a status of Active forevermore!