Kubernetes: Up & Running
Chapter 2 – Creating and Running Containers

Key Points In Chapter Introduction

- Platform for managing distributed applications
- Applications run on individual machines
- Programs accept input manipulate data & return results
- How to build the containers (application container images) that hold these programs.

Nature of Applications

Typically comprised of:

- Language runtimes
- Libraries
- Source code
- Shared components in specific environments

Problems with Dependencies

- Development system dependencies not available on production environment
- Deployment errors in dependency inclusion or configuration
- Differing versions of shared libraries
- Unreliable implementations of deployed applications.

Packaging Applications Into Containers

- Chapter 1 Case for immutable images and infrastructure
- Containers provide this application immutability
- Docker is the default container runtime engine
  - Is that K8s default or technology default?
  - Package an executable
  - Push to Registries
- Docker registries are available on all major public clouds
- You can create your own registries

Containers

- Bundle a program and dependencies into a single artifact
- Most popular format is the Docker format.
- The Docker format has been Standardized by the Open Container Initiative (OCI)
- K8s support both Docker and OCI

Container Images

- Docker image format is defacto standard
- Made up of series of filesystem layers
- Each layer adds, removes or modifies files from preceding layer
- This *Overlay* filesystem is used both in packing and running the application.
- Containers usually include configurations and instructions on how to setup and run the container environment.
- System containers mimic virtual machines and run full boot processes.
- Application containers typically run a single program

Building Docker Application Images

- Docker files are used to automate the creation of an image
- Dockerfile is the recipe for the build
- Includes:
  - Dependencies on other containers and libraries
  - Directory configurations
  - Source code/app files
  - Command to run when starting the container
- Optimization discussion
- Image Security
- Multistage Builds
- Storing in a repository

Docker Image Build Examples

- I was able to recreate the first example with one small tweak in the run command.
- I was not able to successfully recreate the second image build.
- I was able to successfully run the second example from the registry

Addendum

- Portainer.io – Docker management application
  - Helped me figure out my Docker “explorations”
  - Helping me manage my small but “growing to grow” docker images
- Installing Docker of Fedora 31
  - https://linuxconfig.org/how-to-install-docker-on-fedora-31
  - https://www.linuxuprising.com/2019/11/how-to-install-and-use-docker-on-fedora.html
- Setting up a static ip address for a Docker container
  - https://stackoverflow.com/questions/27937185/assign-static-ip-to-docker-container