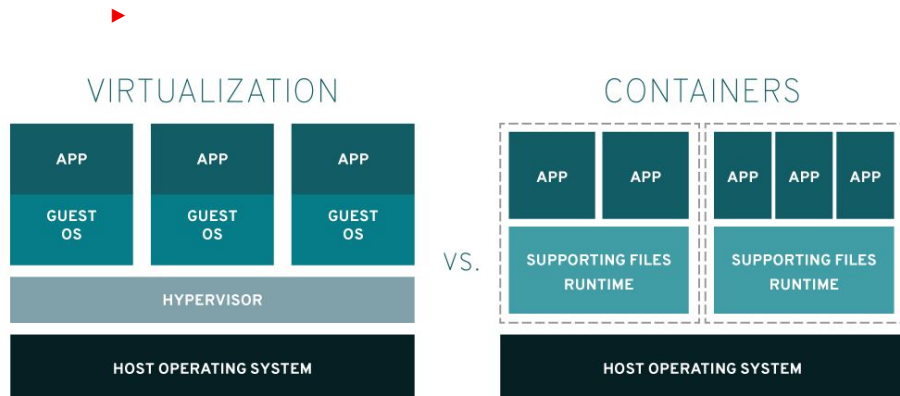


Install Nothing!

Containerizing Your Dev Workflow

Lisa Ranjbar

What is a container?



- ▶ A container is a package that contains all the dependencies needed to run an application and the application itself.
- ▶ When you run an application inside of a container it will have its own:
 - Filesystem
 - Network
 - Memory space

How the magic works

- ▶ Open Container Initiative (OCI): <https://opencontainers.org/>
 - Defines the underlying specifications for containers.
 - Image Specification and Runtime Specification
- ▶ Container Build Tools: Tools that help you build and manage your containers using the OCI Image Specification.
- ▶ Container Runtimes: Libraries that these container tools use to run your containers using the OCI Runtime Specification.

Container Registries

- ▶ A container registry is a place to store containers for future use. There are public and private registries to store your containers.
 - Public: Docker.io and Quay.io
- ▶ When working with public registries you should pick from trusted sources. Docker has the “official” image status for a set of curated images on Docker.io
 - https://hub.docker.com/search?q=&type=image&image_filter=official
- ▶ For Red Hat images you can find them on Quay.io

Building Your First Container

- ▶ Install Podman:
 - <https://podman.io/getting-started/installation>
- ▶ Write a simple application
- ▶ Write a Dockerfile (Containerfile)
 - <https://docs.docker.com/engine/reference/builder/>
- ▶ Build our container.

Picking Your Base Image

What does my application need to run?

- ▶ Programming Language based image
 - Python, Golang, Nodejs, Ruby, etc
 - Good route if you just need the language itself.
- ▶ OS based image
 - Fedora, CentOS, Alpine, Ubuntu, etc
 - Good route if you are going to be installing more development tools into your container like git.

Container Building Demo

1. Build our container

```
podman build -t hello:cli -f Dockerfile.hello .
```

2. Run our container interactively

```
podman run -it --rm hello:cli bash
```

3. Run our container with an entrypoint

```
podman run -it --rm hello:cli --name=Lisa
```

4. Run a containerized web application

```
podman run -it --rm -p 5000:5000 hello:web
```

Introducing Buildah

What if I don't want to rebuild the container every change to the Dockerfile?

- ▶ Installing Buildah:
 - <https://github.com/containers/buildah/blob/main/install.md>
- ▶ Buildah is a command line tool that allows you to build a container interactively.

Introducing Buildah

What if I don't want to rebuild the container every change to the Dockerfile?

- ▶ Let's build our previous container directly with buildah

```
newcontainer=$(buildah from python:3) && echo $newcontainer
```

```
buildah copy $newcontainer .
```

```
buildah config --entrypoint $newcontainer '["python", "hello-cli.py"]'
```

```
buildah commit --rm $newcontainer hello:cli
```

- ▶ Now we can run it with podman like we did before!

```
podman run -it --rm hello:cli --name=Lisa
```

Managing Your Containers

- ▶ Now that we've built our containers locally we can push them to external repositories.
- ▶ Rebuild my local container with the external name:

```
podman build -t quay.io/ltanjabar/hello:cli -f Dockerfile.hello-1 .
```

- ▶ Push my image to the external repository:

```
podman push quay.io/ltanjabar/hello:cli
```

Introducing Skopeo

- ▶ Installing Skopeo: <https://github.com/containers/skopeo/blob/main/install.md>
- ▶ Skopeo is a tool to interact with images in external container repositories.
- ▶ Copy an image to a new tag:

```
skopeo copy docker://quay.io/loranjbar/hello:cli docker://quay.io/loranjbar/hello:latest
```

- ▶ Inspect my external image:

```
skopeo inspect docker://quay.io/loranjbar/hello
```

Fedora Toolbox: A Container Playground

- ▶ Installing Toolbox
- ▶ Toolbox is a tool that will create a containerized command line environment for you. This is useful when you want to try out new tools without installing them on your base operating system.

Containers Are Everywhere

- ▶ While this presentation focused on Linux containers there are containers on just about every operating system these days.
- ▶ The overall workflow I showcased today can be applied in general across the container ecosystem.
- ▶ For those who are on Mac and Windows I still recommend starting out with the Linux ecosystem for containers if at all possible.

More Resources

- ▶ General Containers:
 - [Open Container Initiative](#)
 - [What is a Linux Container?](#)
 - [What is Container Security?](#)
- ▶ Container Orchestration:
 - [What is Kubernetes?](#)

Q&A

Thanks!

Follow me on:

Twitter: @lranjbar

Github: github.com/lranjbar

Linkedin: linkedin.com/in/lranjbar