Next Generation Tools for container technology

Dan Walsh  @rhatdan
Please Stand
Please read out loud all text in RED
I Promise
To say Container Registries rather than Docker registries
I Promise
To say Container Images Rather than Docker images
I Promise
To say Containers Rather than Docker Containers
Sit Down
What do you need to run a container

- Standard Definition of what makes up a container image.
  - OCI Image Bundle Definition
Introducing Skopeo

https://github.com/containers/skopeo

#nobigfatdaemons
Skopeo

- $ skopeo inspect docker://docker.io/fedora
  $ skopeo copy docker://busybox:latest dir:existingemptydirectory
- $ skopeo delete docker://localhost:5000/imagename:latest
What do you need to run a container:

- Standard Definition of what makes up a container image.
  - OCI Image Bundle Definition

- Mechanism to pull images from a container registry to the host
  - github.com/containers/image
What do you need to run a container

- Standard Definition of what makes up a container image.
  - OCI Image Bundle Definition
- Mechanism to pull images from a container registry to the host
  - github.com/containers/image
- Ability to explode images onto COW file systems on disk
  - github.com/containers/storage
What do you need to run a container

- Standard Definition of what makes up a container image.
  - OCI Image Bundle Definition

- Mechanism to pull images from a container registry to the host
  - github.com/containers/image

- Ability to explode images onto COW file systems on disk
  - github.com/containers/storage

- Standard mechanism for running a container
  - OCI Runtime Spec (1.0)
  - runc default implementation of OCI Runtime Spec (Same tool Docker uses to run containers)
OPENSHIFT
#nobigfatdaemons
What does OpenShift/Kubernetes need run a container?

CRI - Container Runtime Interface

#nobigfatdaemons
What does OpenShift/Kubernetes need to run a container?

CRI - Container Runtime Interface

Kubernetes tells CRI to run Container Image:

#nobigfatdaemons
What does OpenShift/Kubernetes need to run a container?

CRI - Container Runtime Interface

Kubernetes tells CRI to run Container Image:

- CRI needs to pull image from Container Registry
What does OpenShift/Kubernetes need to run a container?

CRI - Container Runtime Interface

Kubernetes tells CRI to run Container Image:

- CRI needs to pull image from Container Registry
- CRI Needs to store image on COW File system
What does OpenShift/Kubernetes need to run a container?

Kubernetes tells CRI to run Container Image:

- CRI needs to pull image from Container Registry
- CRI needs to store image on COW File system
- CRI Needs to execute OCI Runtime
Introducing CRI-O

#nobigfatdaemon
Introducing CRI-O

CRI-O - OCI-based implementation of Kubernetes Container Runtime Interface

- Scope tied to kubernetes CRI
- Only supported user is kubernetes
- Uses standard components as building blocks

“Nothing more, Nothing Less”
MESOSPHERE
#nobigfatdaemons
Overview of additional components

- **oci-runtime-tools** library is used to generate OCI configs for containers
Overview of additional components

- **oci-runtime-tools** library is used to generate OCI configs for containers
- **CNI** is used for setting up networking
  - Tested with Flannel, Weave and openshift-sdn

#nobigfatdaemons
Overview of additional components

- **oci-runtime-tools** library is used to generate OCI configs for containers
- **CNI** is used for setting up networking
  - Tested with Flannel, Weave and openshift-sdn
- **conmon** is a utility for:
  - Monitoring
  - Logging
  - Handling tty
  - Serving attach clients
  - Detecting and reporting OOM
Pod architecture (runc)

Pod
(ipc, net, pid namespaces, cgroups)

Conmon

Infra Container

Container A (runc)

Container B (runc)

#nobigfatdaemons
Pod architecture (Kata Containers)

- **Virtual Machine**
  - (ipc, net, pid namespaces, cgroups)

- **Pod**
  - (net namespace, cgroups)

- **Container A**
  - (kata-runtime)

- **Container B**
  - (kata-runtime)

- **kata-shim**
  - Conmon

- **Pod**
  - #nobigfatdaemons
# Architecture

[Diagram of container architecture with pod 1 and pod 2, each containing container A, B, and C, connected to common interfaces, and linked to kubectl, gRPC, and library services with GitHub repositories for image and storage.]
Status

- All e2e, cri-tools, integration, 9 test suites, (>500) tests passing.
  - No PRs merged without passing all the tests
Status

- **All** e2e, cri-tools, integration, 9 test suites, (>500) tests passing.
  - No PRs merged without passing all the tests.
- 1.0.7 (kube 1.7.x) supported. (December 2017)
Status

- **All e2e, cri-tools, integration, 9 test suites, (>500) tests passing.**
  - No PRs merged without passing all the tests.
- 1.0.7 (kube 1.7.x) supported. (December 2017)
- 1.9.12 (kube 1.9.x) released.
  - CRI-O fully supported in OpenShift 3.9 along with docker.
Status

- **All** e2e, cri-tools, integration, 9 test suites, (>500) tests passing.
  - No PRs merged without passing all the tests.
- 1.0.7 (kube 1.7.x) supported. (December 2017)
- 1.9.12 (kube 1.9.x) released.
  - CRI-O fully supported in OpenShift 3.9 along with docker.
- 1.10.6 (kube 1.10.x) released.
Status

- All e2e, cri-tools, integration, 9 test suites, (>500) tests passing.
  - No PRs merged without passing all the tests.
- 1.0.7 (kube 1.7.x) supported. (December 2017)
- 1.9.12 (kube 1.9.x) released.
  - CRI-O fully supported in OpenShift 3.9 along with docker.
- 1.10.6 (kube 1.10.x) released.
- 1.11.2 (Kube 1.11.x) released
Status

- **All** e2e, cri-tools, integration, 9 test suites, (>500) tests passing.
  - No PRs merged without passing all the tests.
- 1.0.7 (kube 1.7.x) supported. (December 2017)
- 1.9.12 (kube 1.9.x) released.
  - CRI-O fully supported in OpenShift 3.9 along with docker.
- 1.10.6 (kube 1.10.x) released.
- 1.11.2 (Kube 1.11.x) released
- 1.12.1 (Kube 1.12.x) released
- Goal for Openshift 4.0 is to fully support CRI-O by default.
CRI-O is now powering nodes on OpenShift Online.
"CRI-O just works for them, so they haven’t had much to say"
Making running containers in production

boring

#nobigfatdaemons
What else does OpenShift need?

- Ability to build container images
- Ability to push container images to container registries
Introducing Buildah

https://github.com/containers/buildah
Coreutils for building containers. Simple interface
Coreutils for building containers. Simple interface
# ctr=$(buildah from fedora)
Coreutils for building containers. Simple interface
# ctr=$(buildah from fedora)
# mnt=$(buildah mount $ctr)
# docker cp

**Estimated reading time:** 5 minutes

## Description

Copy file/folders between a container and the local filesystem

## Usage

```
docker cp [OPTIONS] CONTAINER:SRC_PATH DEST_PATH
```

```
docker cp [OPTIONS] SRC_PATH: CONTAINER:DEST_PATH
```

## Options

<table>
<thead>
<tr>
<th>Name, shorthand</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>--archive, -a</td>
<td></td>
<td>Archive mode (copy all and gid information)</td>
</tr>
<tr>
<td>--follow-link, -L</td>
<td></td>
<td>Always follow symbol link in SRC_PATH</td>
</tr>
</tbody>
</table>

## Parent command

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>docker</td>
<td>The base command for the Docker CLI.</td>
</tr>
</tbody>
</table>

## Extended description

The `docker cp` utility copies the contents of `SRC_PATH` to the `DEST_PATH`. You can copy from the container's file system to the local machine or the reverse, from the local filesystem to the container. If `--` is specified for either the `SRC_PATH` or `DEST_PATH`, you can also stream a tar archive from `STDIN` or to `STDOUT`. The `CONTAINER` can be a running or stopped container. The `SRC_PATH` or `DEST_PATH` can be a file or directory.
Coreutils for building containers. Simple interface

```
# ctr=$(buildah from fedora)
# mnt=$(buildah mount $ctr)
# cp -R src $mnt
```
Coreutils for building containers. Simple interface
# ctr=$(buildah from fedora)
# mnt=$(buildah mount $ctr)
# cp -R src $mnt
# dnf install --installroot=$mnt httpd
Coreutils for building containers. Simple interface

# ctr=$(buildah from fedora)
# mnt=$(buildah mount $ctr)
# cp -R src $mnt
# dnf install --installroot=$mnt httpd
# make install DESTDIR=$mnt
Coreutils for building containers. Simple interface

```bash
# ctr=$(buildah from fedora)
# mnt=$(buildah mount $ctr)
# cp -R src $mnt
# dnf install --installroot=$mnt httpd
# make install DESTDIR=$mnt
# buildah config --entrypoint=/usr/sbin/test.sh --env foo=bar $ctr
```

#nobigfatdaemons
Coreutils for building containers. Simple interface
# ctr=$(buildah from fedora)
# mnt=$(buildah mount $ctr)
# cp -R src $mnt
# dnf install --installroot=$mnt httpd
# make install DESTDIR=$mnt
# buildah config --entrypoint=/usr/sbin/test.sh --env foo=bar $ctr
# buildah commit $ctr myhttpd
Coreutils for building containers. Simple interface

# ctr=$(buildah from fedora)
# mnt=$(buildah mount $ctr)
# cp -R src $mnt
# dnf install --installroot=$mnt httpd
# make install DESTDIR=$mnt
# buildah config --entrypoint=/usr/sbin/test.sh --env foo=bar $ctr
# buildah commit $ctr myhttpd
# buildah push myhttpd docker://rhatdan/myhttpd
Dan Wait!

#nobigfatdaemons
Dan Wait!
What about Dockerfile??????

#nobigfatdaemons
Buildah also supports Dockerfile
buildah build-using-dockerfile -f Dockerfile .
Buildah also supports Dockerfile
buildah build-using-dockerfile -f Dockerfile .
Or for those lazy ones:
buildah bud -f Dockerfile .
Does Buildah have a scripting language? Perhaps Buildahfile?
BASH
We want others to build higher level tools on Buildah.
We want others to build higher level tools on Buildah.
Working to make OpenShift use Buildah for S2I containers rather than use Docker.
BASH

We want others to build higher level tools on Buildah.
Working to make OpenShift use Buildah for S2I containers rather then use Docker.
Want to work with Ansible-containers to use buildah for containers as well.
What else does OpenShift need?

- Ability to diagnose problems on the host
- If you don’t use Docker to run the containers, how does an admin discover what is going on in his Container runtime, without the docker CLI?
Introducing podman
part of the libpod effort

#nobigfatdaemons
Introducing podman part of the libpod effort

podman is tool for managing POD/Containers based on the Docker CLI

https://github.com/containers/libpod
Introducing podman

podman is tool for managing POD/Containers based on the Docker CLI

# podman ps -a

https://github.com/containers/libpod
Introducing podman

podman is tool for managing POD/Containers based on the Docker CLI

### podman ps -a

### podman run -ti fedora sleep 2000

https://github.com/containers/libpod

#nobigfatdaemons
Introducing podman

podman is tool for managing POD/Containers based on the Docker CLI

# podman ps -a

# podman run -ti fedora sleep 2000

# podman exec -ti fedora sh

https://github.com/containers/libpod

#nobigfatdaemons
Introducing podman

podman is tool for managing POD/Containers based on the Docker CLI

# podman ps -a

# podman run -ti fedora sleep 2000

# podman exec -ti fedora sh

# podman images

...

https://github.com/containers/libpod
Alan Moran on Twitter: "I completely forgot that ~2 months ago I set up "alias docker=podman" and it has been a dream. #nobigfatdaemons @projectatomic"
Building Containers the hard and easy way
Saturday August 18, 2018 3:50pm - 4:25pm

Container Security: So many options, use them all
Saturday August 18, 2018 2:10pm - 2:45pm
Questions

Blog: https://medium.com/cri-o

Github:
- https://github.com/kubernetes-sigs/cri-o
- https://github.com/containers/buildah
- https://github.com/containers/skopeo
- https://github.com/containers/libpod (podman)
- https://github.com/containers/storage
- https://github.com/containers/image

Site: https://cri-o.io IRC: freenode: #cri-o
Site: https://podman.io IRC: freenode: #podman
Site: https://buildah.io IRC: freenode: #buildah