



**Red Hat**

# Forensic Container Checkpointing and Analysis

Adrian Reber  
Senior Principal Software Engineer

FOSDEM 2024, February 03

# Agenda

Background

Use cases

Forensic Container Analysis

Future

# Checkpoint/Restore in Userspace

## CRIU

# Multiple Integrations Exist

# Container Live Migration

## OpenVZ

# Container Live Migration

## Borg

# Container Live Migration

## LXC/LXD

# Container Live Migration

## Docker



# Container Live Migration

## Podman

# Container Live Migration

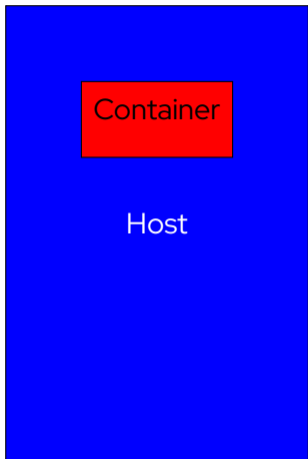
## CRI-O

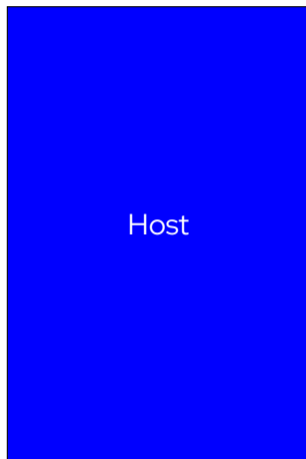
# Forensic Container Checkpointing

- <https://github.com/kubernetes/enhancements/pull/1990>
- <https://github.com/kubernetes/enhancements/pull/3264>
- <https://github.com/kubernetes/kubernetes/pull/104907>
- <https://kubernetes.io/blog/2022/12/05/forensic-container-checkpointing-alpha/>
- <https://kubernetes.io/blog/2023/03/10/forensic-container-analysis/>
- <https://www.opensourcerers.org/2023/09/11/forensic-container-checkpointing-in-openshift>

# Use Cases

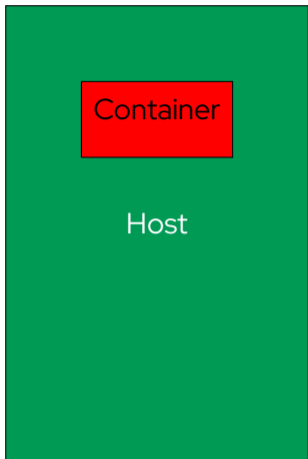
# Reboot and Save State



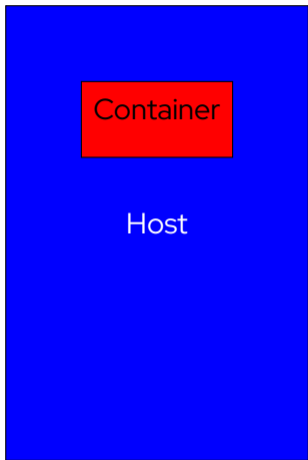


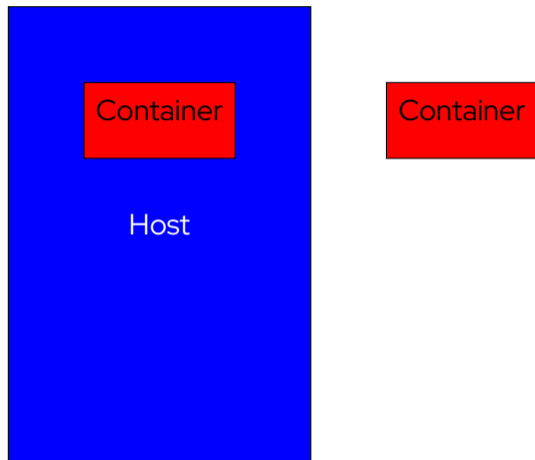
Container

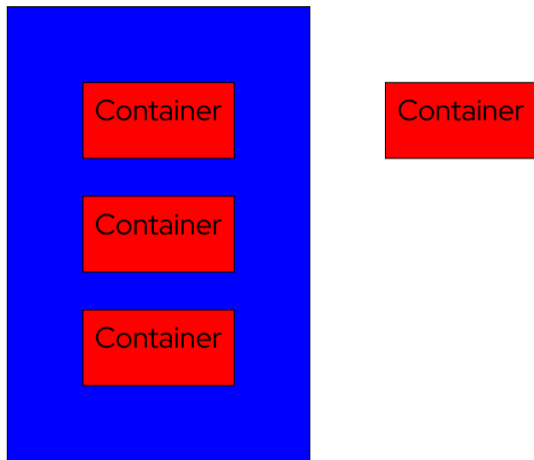




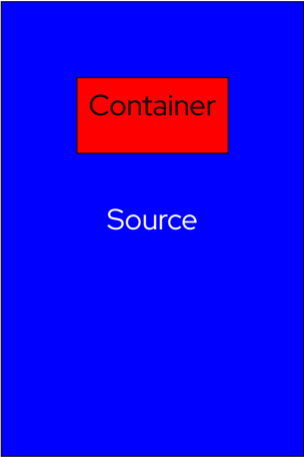
# Quick Startup

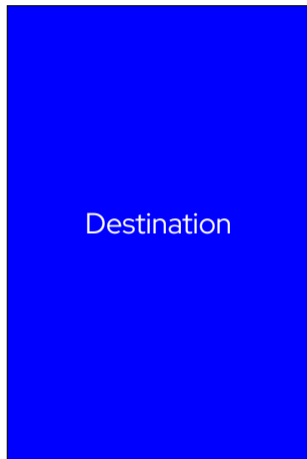
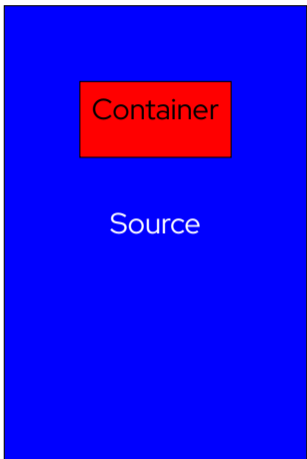




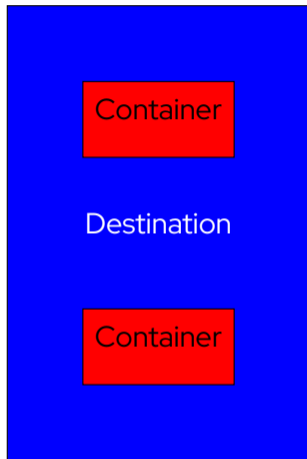
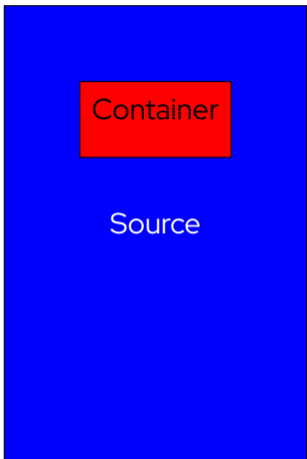


# Container Live Migration









# Spot instances

# Forensic Container Analysis

# Forensic Container Analysis

## Stateful Container



# Forensic Container Analysis

```
checkpointctl
```

# Forensic Container Analysis

`crit`

# Forensic Container Analysis

grep

# Forensic Container Analysis

`gdb`



What's next?

kubectl checkpoint

# Pod Checkpoint/Restore

```
kubectl migrate
```

# Scheduler Integration

# Checkpoint Image Standard

<https://github.com/opencontainers/image-spec/issues/962>

# Summary

- CRIU can checkpoint and restore containers
- Integrated in different containers engines
- Used in production
- Reboot into new kernel without losing container state
- Start multiple copies
- Migrate running containers
- Spot instances
- Forensic container checkpointing (KEP #2008)



# Thank you