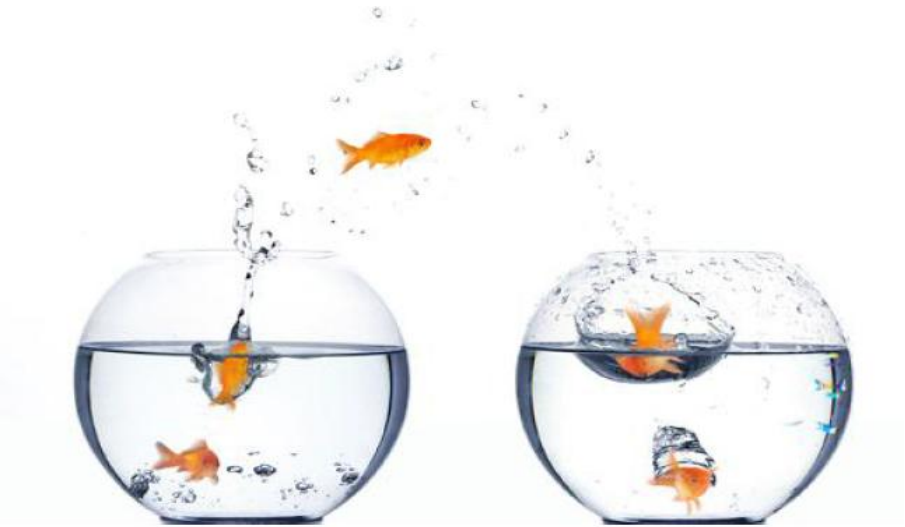
Docker容器热迁移

Author/ Email:
李泽帆/lizefan@huawei.com
邓广兴/dengguangxing@huawei.com

关于热迁移

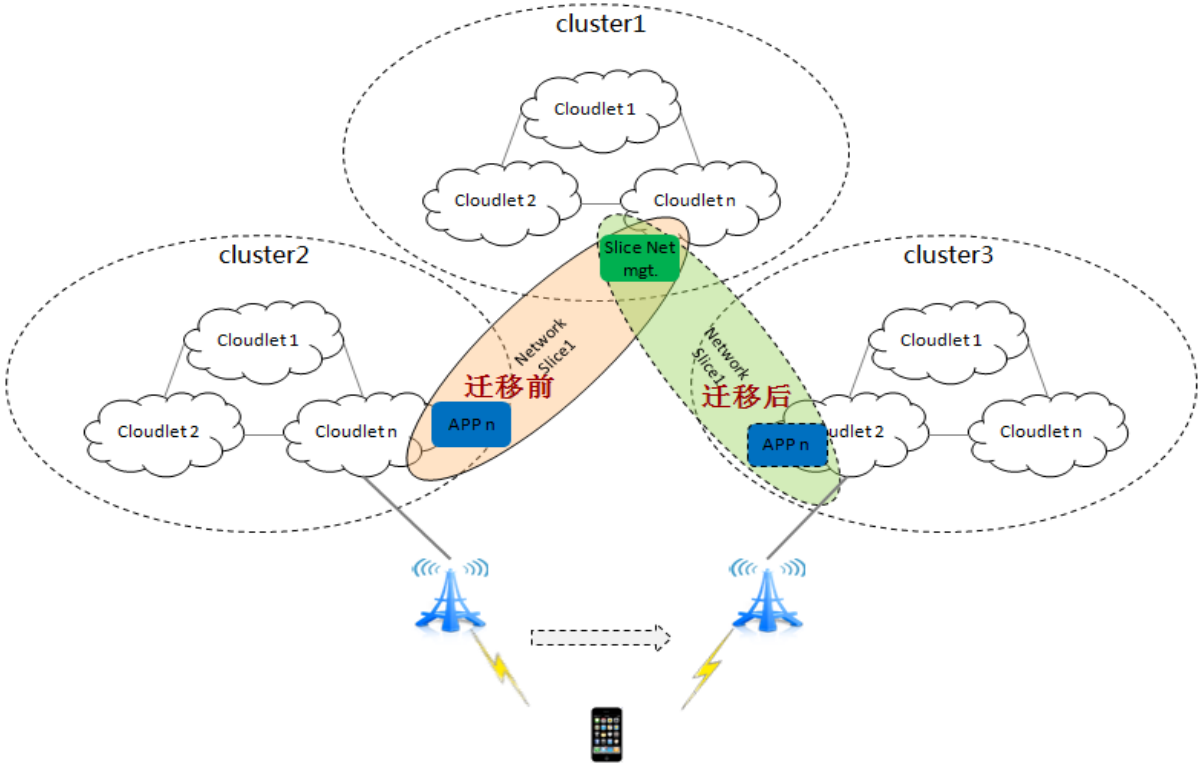
概念



- 主体：处于运行状态的虚拟机或应用程序
- 动作：在不同主机之间进行迁移
- 要求：内存、存储和网络连接等状态能够保持

[–from wikipedia]

场景


cluster1
Cloudlet 1
Cloudlet 2
Cloudlet n
Slice Net mgt.
cluster2
Cloudlet 1
Cloudlet 2
Cloudlet n
Network Slice1
迁移前
APP n
Network Slice1
迁移后
cluster3
Cloudlet 1
Cloudlet 2
Cloudlet n
APP n

热迁移现状

热迁移现状

- 虚拟机热迁移：
  - 几乎所有虚拟化方案都已实现热迁移功能
    - KVM，Vmware，Hyper-V，Xen … …
- 容器热迁移：
  - 没有成熟的方案
  - 容器迁移≈进程迁移

关于CRIU

进程迁移工具-CRIU

- CRIU – Checkpoint/Restore In Userspace
- 用途：
  - 将进程状态保存为文件
  - 通过文件恢复进程
- 使用方式：
  - criu dump|pre-dump –t $PID –images-dir=xx … …
  - criu restore –images-dir=xx … …
  - Criu page-server

- 主页：https://criu.org/Main_Page
- 代码：https://github.com/xemul/criu

CRIU的历史背景

- Kernel-based Checkpoint and Restart
  - 作者：Oren Laadan，哥伦比亚大学博士
  - 时间：2007年发表论文，2008年发表开源实现
  - 实现：100个patch，修改了几十个内核子系统

- Checkpoint Restore In Userspace（CRIU）
  - 作者：Pavel Emelyanov等，Paralles公司
  - 时间：2011年开始
  - 实现：累计180+个patch进入主线

> “This is a project by various mad Russians to perform c/r mainly from userspace... However I'm less confident than the developers that it will all eventually work!”
>
> - Andrew Morton (2012-01)

CRIU的缺陷

- 虚拟机
  - 整体的内存拷贝迁移
  - 有限的设备状态保存恢复
- 容器
  - 进程的用户态内存
  - 进程保存在内核的各种状态
    - virtual memory mappings
    - open files
    - credentials
    - timers
    - process ID
    - ...
- CRIU不支持的应用
  - 使用了以下特性
    - Tasks with debugger attached
    - Task running in compat mode
    - UNIX sockets with relative path
    - Sockets other than TCP, UCP, UNIX, packet and netlink
    - Cork-ed UDP sockets
    - SysVIPC memory segment without IPC namespace
    - ...
  - 图形应用
  - 打开、内存映射字符或块设备

"This is not an enterprise feature. It's a promise one cannot keep. We will not add code to systemd that works often but not always, and CRIU is certainly of that kind."

- Lennart Pottering (systemd-devel, 2015)

容器热迁移现有方案

Docker的支持：runc/libcontainer

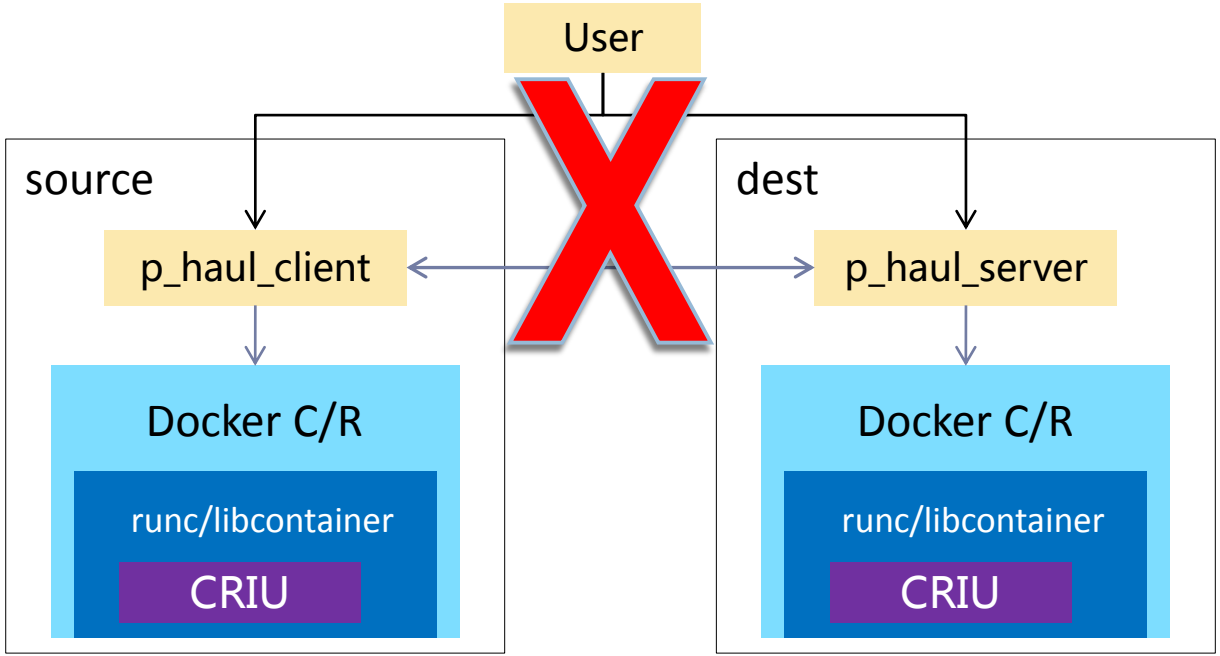
- 基于CRIU实现两个C/R相关接口：
  - checkpoint
  - restore

- 调用方式：
  - runc命令行
  - libcontainer接口函数

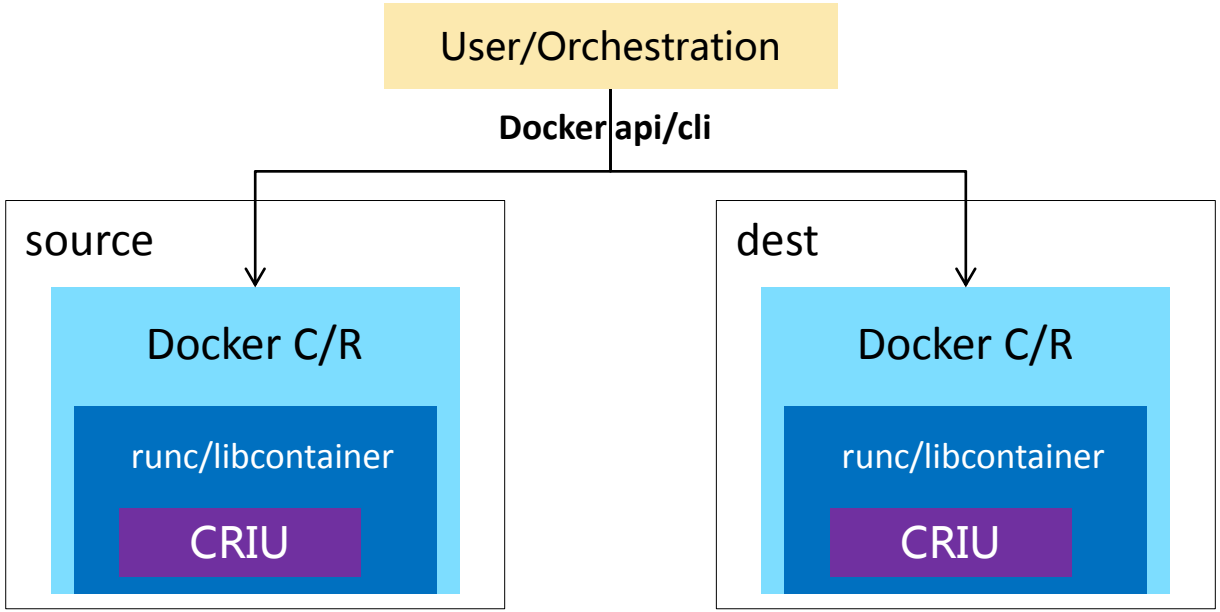
Docker的支持： C/R

- PR：
  - https://github.com/docker/docker/pull/13602 (closed)
- 提供两个命令：
  - docker checkpoint $CONTAINER
  - docker restore $CONTAINER
- 限制
  - 本地容器状态保存和恢复，无法跨主机迁移

业界方案：P.HAUL

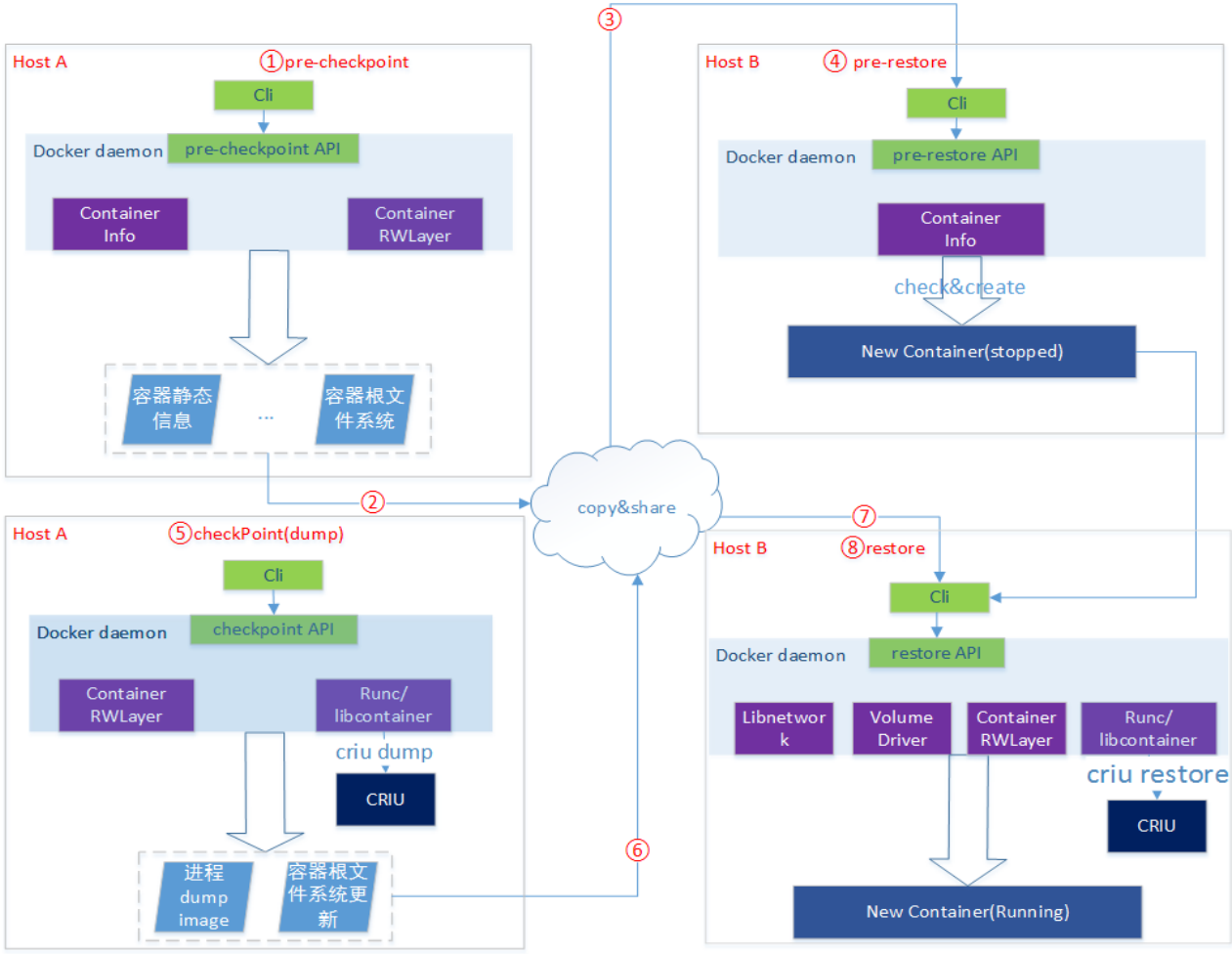

User
source
dest
p_haul_client
p_haul_server
Docker C/R
runc/libcontainer
CRIU
Docker C/R
runc/libcontainer
CRIU

我们希望的模式：


User/Orchestration
Docker api/cli
source
Docker C/R
runc/libcontainer
CRIU
dest
Docker C/R
runc/libcontainer
CRIU

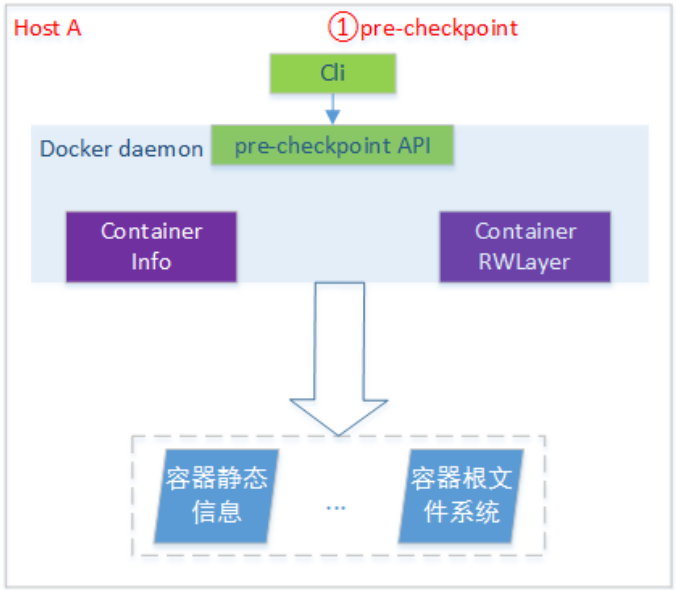
华为容器热迁移实现

流程


Host A
①pre-checkpoint
Cli
Docker daemon
pre-checkpoint API
Container Info
Container RWLayer
容器静态信息
...
容器根文件系统
③
Host B
④ pre-restore
Cli
Docker daemon
pre-restore API
Container Info
check&create
New Container(stopped)
②
copy&share
⑦
Host A
⑤checkPoint(dump)
Cli
Docker daemon
checkpoint API
Container RWLayer
Runc/ libcontainer
criu dump
CRIU
进程 dump image
容器根文件系统更新
⑥
Host B
⑧restore
Cli
Docker daemon
restore API
Libnetwork
Volume Driver
Container RWLayer
Runc/ libcontainer
criu restore
CRIU
New Container(Running)

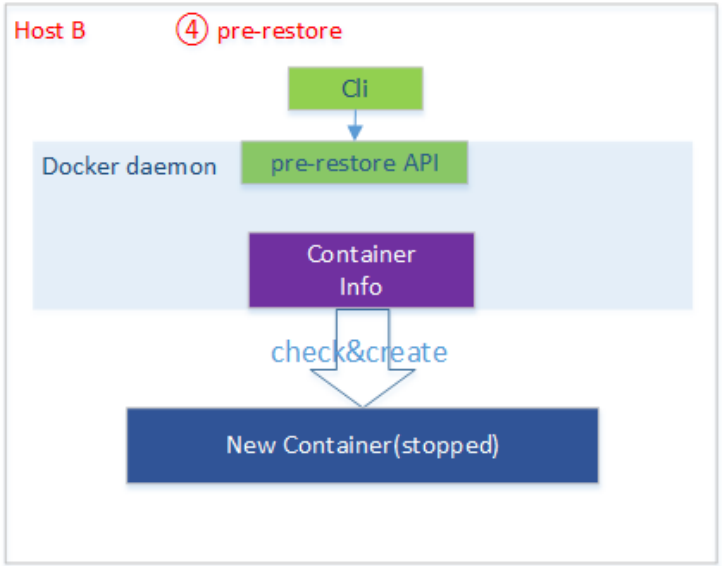
第一步：pre-checkpoint

- 操作：
  - 验证checkpoint可行性
  - 保存容器基础状态文件：
    - 容器静态配置
    - 容器根文件系统
  - 将文件同步到目的主机
    - rsync



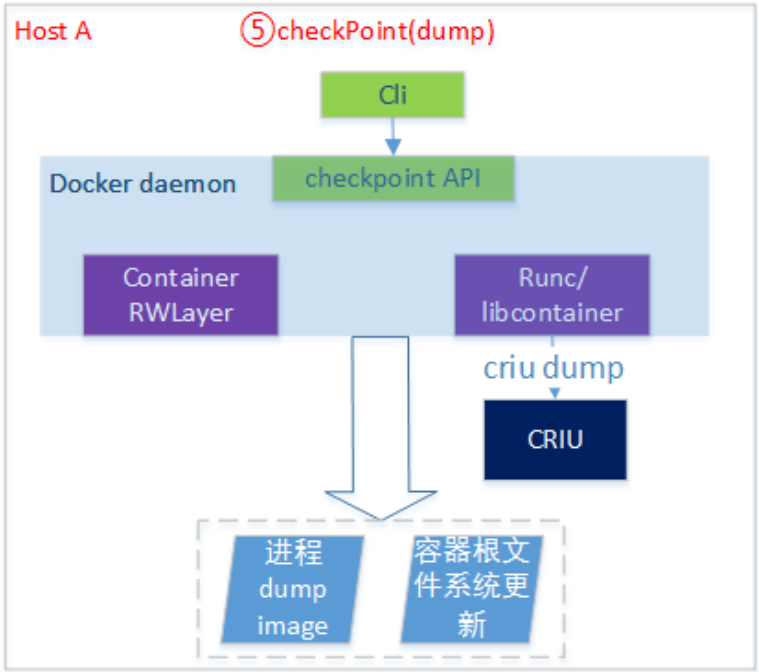
第二步： pre-restore

- 操作：
  - 验证restore可行性
  - 根据配置文件创建容器



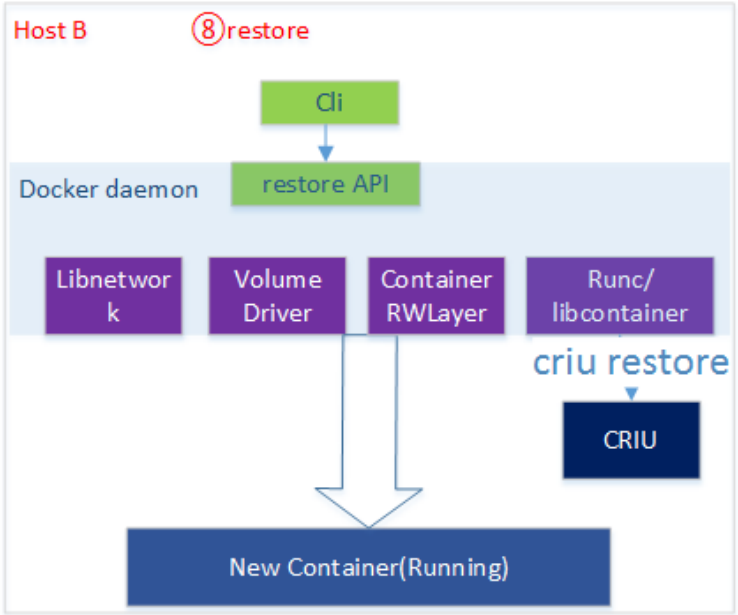
第三步：checkpoint

- 操作：
  - 保存容器进程运行状态：
    - 普通迁移 – criu dump
    - 迭代迁移 – criu pre-dump
  - 增量备份根文件系统
  - 将文件同步到目的主机



第四步：restore

- 操作：
  - 恢复容器根文件系统
  - Libnetwork恢复网络
  - Volume Driver恢复数据卷
  - 恢复容器进程 – criu restore



补充：数据卷迁移

- 本地数据卷
  - 不迁移
- Elara
  - Docker 数据卷插件
  - 跨主机数据卷管理
  - 支持多种后端：SAN, NAS, Cinder等

补充：网络迁移

- CRIU网络方案
  - 由CRIU保存并恢复容器网络环境
- libnetwork方案：
  - 由libnetwork处理网络sandbox和endpoint的迁移和恢复

补充：迭代迁移

- 基于：
  - 内核mem-track机制
  - CRIU pre-dump命令
- 新增接口：
  - runc/libcontainer pre-dump/page-server(*)
- 迭代条件：
  - 最大迭代数
  - 最小脏页数
  - 最大脏页增长率

补充：pod迁移

- pod：一组共享特定Namspace的容器
- CRIU
  - restore： --join-ns NS:PID|NS_FILE
  - 恢复进程时加入指定的Namespace
- Docker
  - restore: --join-ns NS:CONTAINER

现状&待改进

- CRIU本身功能不完善
- 网络迁移进行中
- 根文件系统(rootfs)迁移限制
  - 虚拟机：共享存储，无需拷贝
  - 容器：本地存储，拷贝迁移
  - 问题：rootfs改动较大导致迁移时间变长
  - 方案：使用共享存储管理容器rootfs（Docker目前不支持）
- 同步方式：page-server

《Docker进阶与实战》

机械工业出版社出版

最全面的内容

进阶与实战的最佳选择

华为Docker实践小组 出品

Docker社区贡献全球排名前三、国内排名第一

JD.京东.COM　亚马逊amazon.cn　当当dangdang.com

【团队介绍】

我们致力于打造未来ICT领域的操作系统、云平台，我们致力于研究未来ICT、NFV/SDN、云服务场景下的OS、容器、虚拟化前沿技术。

这里有Linux kernel、Docker社区多名maintainer，有APCI标准提案committer，有精通Linux kernel各子系统（调度、内存管理、文件系统、网络等）的资深专家，有畅销书《Docker进阶与实战》的作者团队和您一起探讨容器虚拟化的未来技术。

如果你相信技术可以改变生活、改变世界；如果你喜欢Linux、喜欢钻研技术；如果你热爱开源、想成为技术大咖；别犹豫，加入我们！

Docker/容器虚拟化架构师/高级工程师（北京/杭州）

【职责】

- 负责x86、arm 64架构下容器虚拟化需求分析、关键技术和特性的设计；
- 负责Docker/容器关键技术研究和特性开发；
- 负责Docker及相关开源社区互动、运作，如推送bug fix和新特性。

【要求】

- 计算机相关专业本科及以上学历，4年以上Linux系统或内核开发经验；
- 熟悉Linux容器相关技术（namespace、cgroup）者优先；
- 熟悉Docker源码、有Docker社区开发经验者优先；
- 熟悉容器编排/调度（Kurbernetes，Mesos等）技术者优先；

【招贤纳士】【华为-Docker团队】
诚邀容器、虚拟化领域专家

【电话】18501294585/北京、13732261657/杭州
【邮箱】hr.kernel@huawei.com