华为云联合openEuler构建全栈轻量安全容器解决方案实践

张天阳/刘昊

开放原子开源基金会 OPENATOM FOUNDATION | OpenEuler

Kuasar：面向下一代的容器运行时

痛 点

云原生场景要求更高

单一沙箱无法同时满足，需要的沙箱容器种类多样

运维压力显著上升

太多的沙箱容器需要运维，复杂程度高、压力大

平滑演进路径缺失

沙箱演进成本高，难以拥抱新沙箱的出现

KUASAR

https://github.com/kuasar-io/kuasar

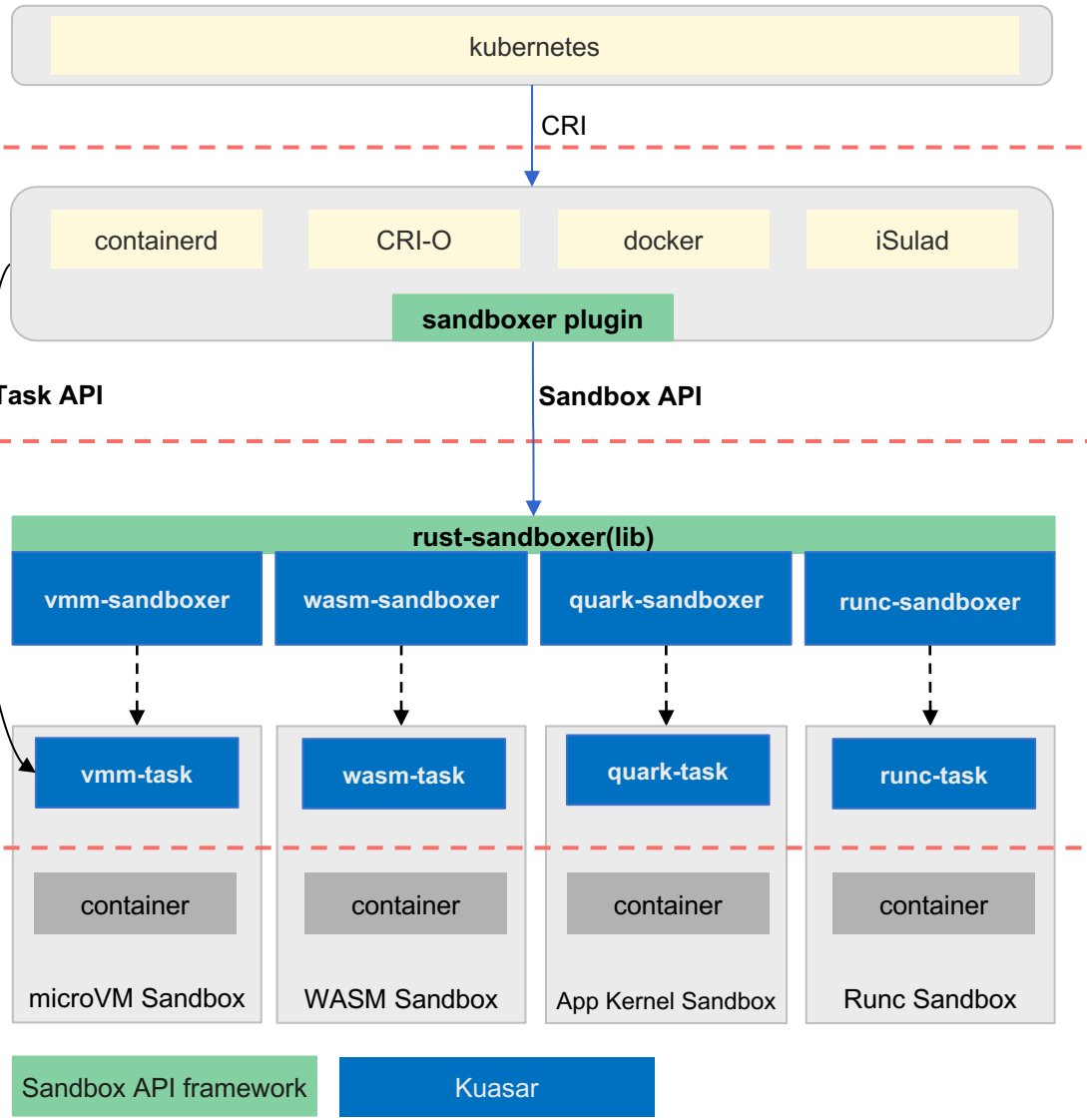
统一沙箱抽象定义	多沙箱混部	极致框架优化
沙箱的定义统一，管理逻辑变得清晰，运维效率得以提高	**内置**集成多种主流沙箱容器，打造开箱即用的容器运行时 安全、隔离、加速、高效	简化模型链路，完全Rust化 **2x** Pod启动速度 **99%** ↓↓↓ 管理进程开销 (100 MicroVM pods)

MicroVM Sandboxer　Wasm Sandboxer　App Kernel Sandboxer　runC Sandboxer

开放中立　多家单位联合发起，目标成为 CNCF 项目

中国农业银行　OpenEuler　HUAWEI　WasmEdgeRuntime　Quark Containers

Kuasar 架构



- **容器编排工具**

  自动化容器的部署、管理、扩展和联网。如 Kubernetes

- **高阶容器引擎**

  负责 CRI 的实现，从高维度管理容器和镜像实例。如 containerd/isulad…

- **低阶容器运行时**

  创建 OCI 标准容器。如 kata/runc/runwasi…

  Kuasar 属于低阶容器运行时，主要包括 2 个模块（可独立，可合并）：

  - **Kuasar-sandboxer：**负责管理沙箱生命周期和资源分配
  - **Kuasar-Task：**负责管理容器的生命周期和资源分配

- **沙箱**

  为容器进程提供安全隔离的环境

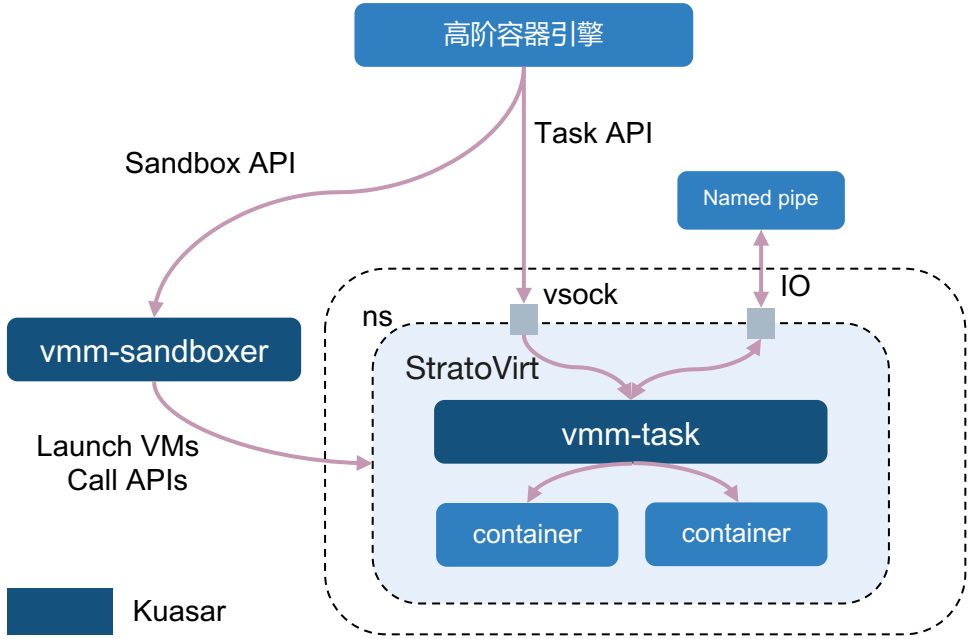
  - **microVM Sandbox**: **Cloud Hypervisor**, Firecracker, **StratoVirt**, **QEMU**
  - **WASM Sandbox**: **WasmEdge, Wasmtime**
  - **App Kernel Sandbox**: gVisor, **Quark**
  - **Runc Sandbox: runC**

Kuasar + StratoVirt 轻量级虚机安全容器

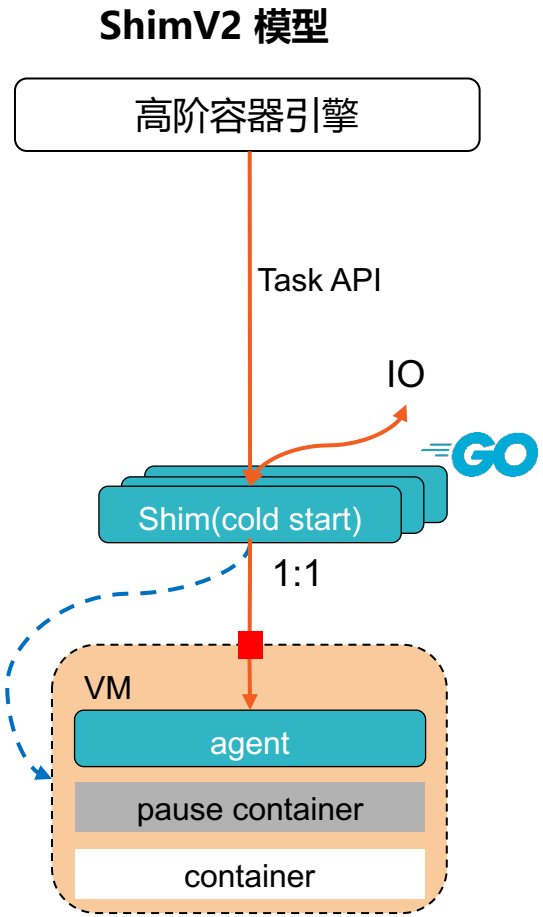
轻量级虚机安全容器将虚拟化技术和容器技术有机结合，可以实现比普通 Linux 容器更好的隔离性。通常由一个轻量级虚机进程提供一套完整的操作系统，并在其中运行容器进程。Kuasar 支持创建以 StratoVirt 为沙箱的安全容器，并与高阶容器引擎对接：

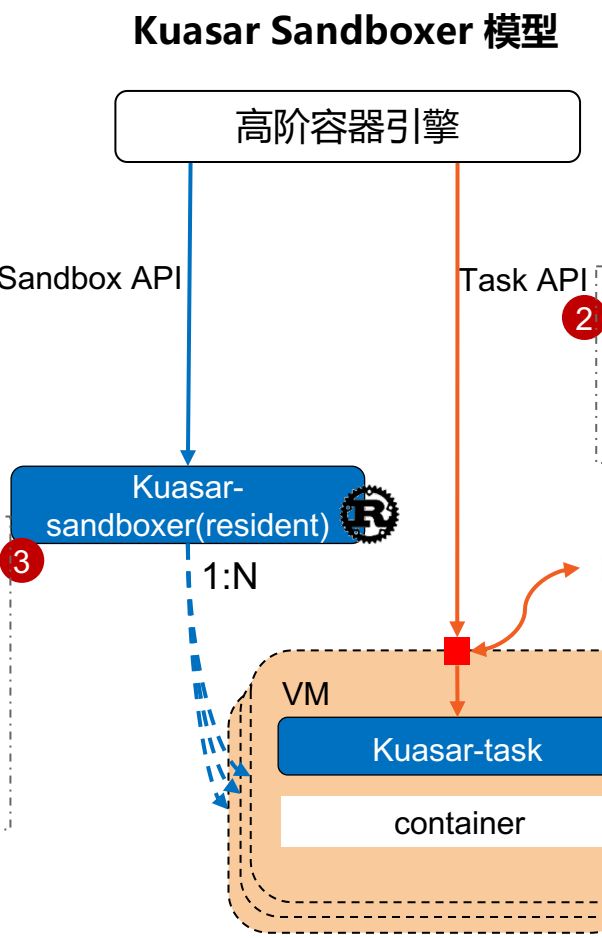
- 沙箱启动：容器引擎通过 Sandbox API 调用 Kuasar 的 vmm-sandboxer 进程，该进程负责拉起一个 StratoVirt 虚机并调用其 API 完成初始化，最后返回 StratoVirt 虚机里的 1 号进程（Kuasar 的 vmm-task 进程）监听的 vsock 地址；

- 容器启动：容器引擎直接通过 vsock 地址调用 vmm-task 的 Task API 接口运行容器，容器的 IO 也由此导出。



Kuasar Sandboxer vs ShimV2 模型





sandbox 管理逻辑清晰 1

sandbox 管理逻辑和 container 管理逻辑完全分开，开发友好，语义清晰

高效的 sandboxer 进程 3

a. Sandboxer 进程常驻减掉了冷启动 Shim 进程的耗时，Pod 启动速度加快

b. 1:N 管理模型大幅减少了进程数量

c. Rust 程序内存安全，相比 Golang 开销小

简化 container 调用链 2

取消 Task API 到 Shim v2 API 的转化，直接调用，链路简化

去除 pause 容器

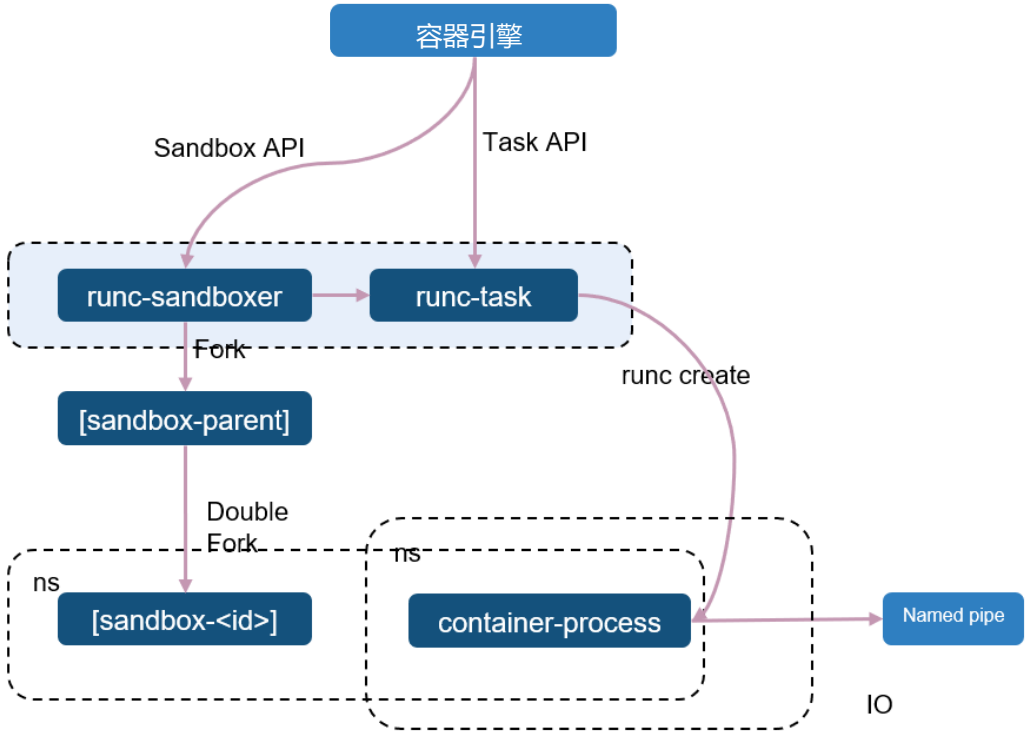
创建 Pod 不再创建 pause 容器，不再需要准备 pause 容器镜像快照，Pod 启动速度加快 4

Kuasar 创建其他类型沙箱容器

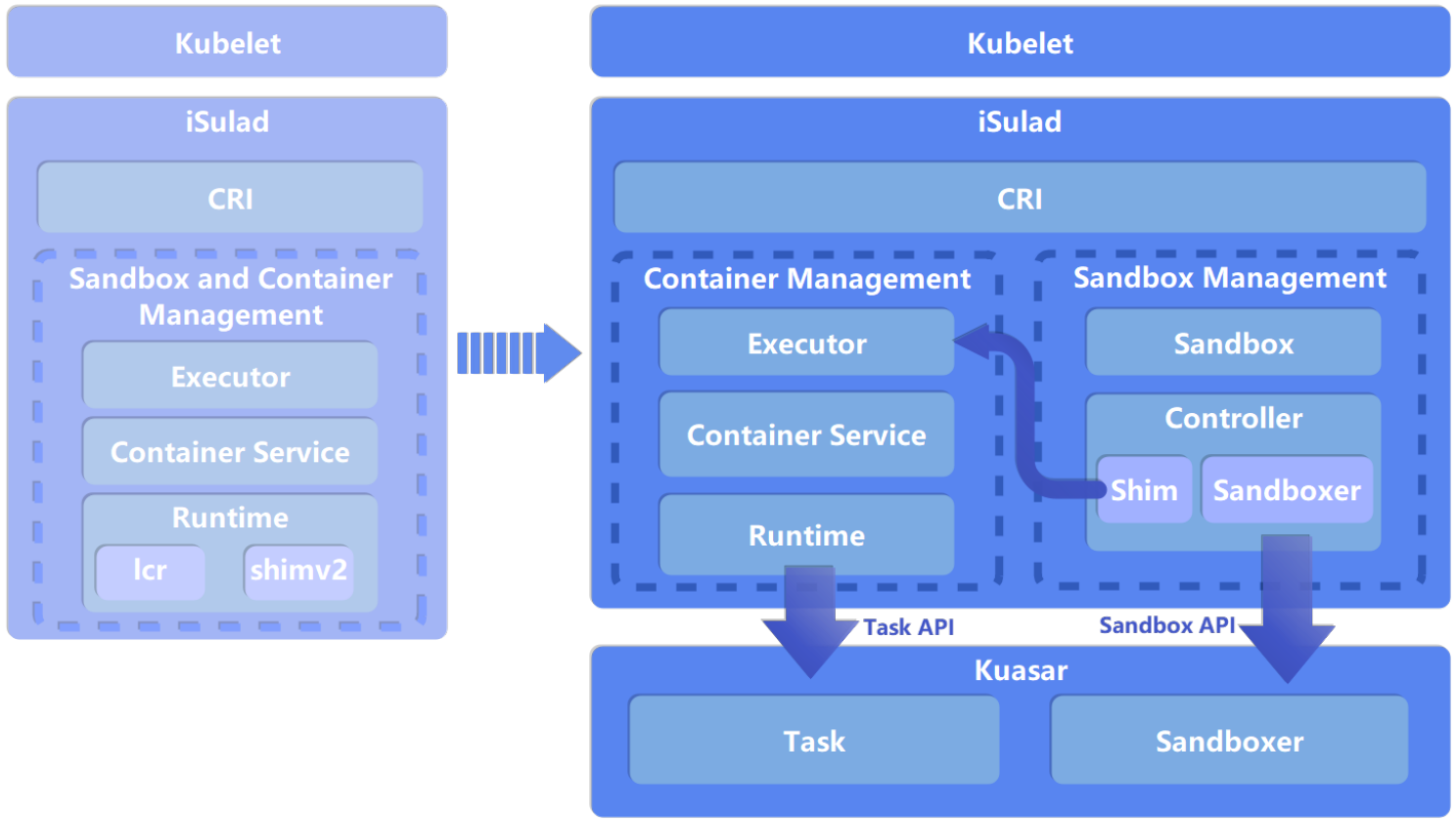
除了支持基于轻量级虚拟化技术的安全容器沙箱，通过 iSulad + Kuasar 还支持基于新兴的 WebAssembly 沙箱、基于进程级虚拟化的 App Kernel 沙箱。在今年12月的 Kuasar v0.4.0 版本中，正式支持基于内核的原生**普通容器沙箱 runC**，并去掉 pause 容器 和 shim 进程：

- 沙箱启动：Kuasar 的 runc-sandboxer 进程 fork 出一个超轻 (<100kB) 的 [sandbox-parent] 进程，为创建容器的 namespace 做准备。

- 容器启动：[sandbox-parent] 进程 fork 一个 [sandbox-<id>] 进程，用于承载容器的 pid/net/ipc/uts/mount ns…
  - 若容器需要共享 pid ns， [sandbox-<id>] 进程常驻，Kuasar 的 runc-task 进程拉起容器进程，并将其加入到该 ns。
  - 如果容器不需要共享 pid ns，[sandbox-<id>] 进程立刻退出，容器进程创建属于自己独立的 pid ns。



iSulad + Kuasar



iSulad 新架构

- *新增沙箱语义*，使沙箱成为容器引擎的一等公民
- 沙箱管理与容器管理解耦，沙箱的管理不再需要通过 OCI Runtime 标准容器接口进行管理

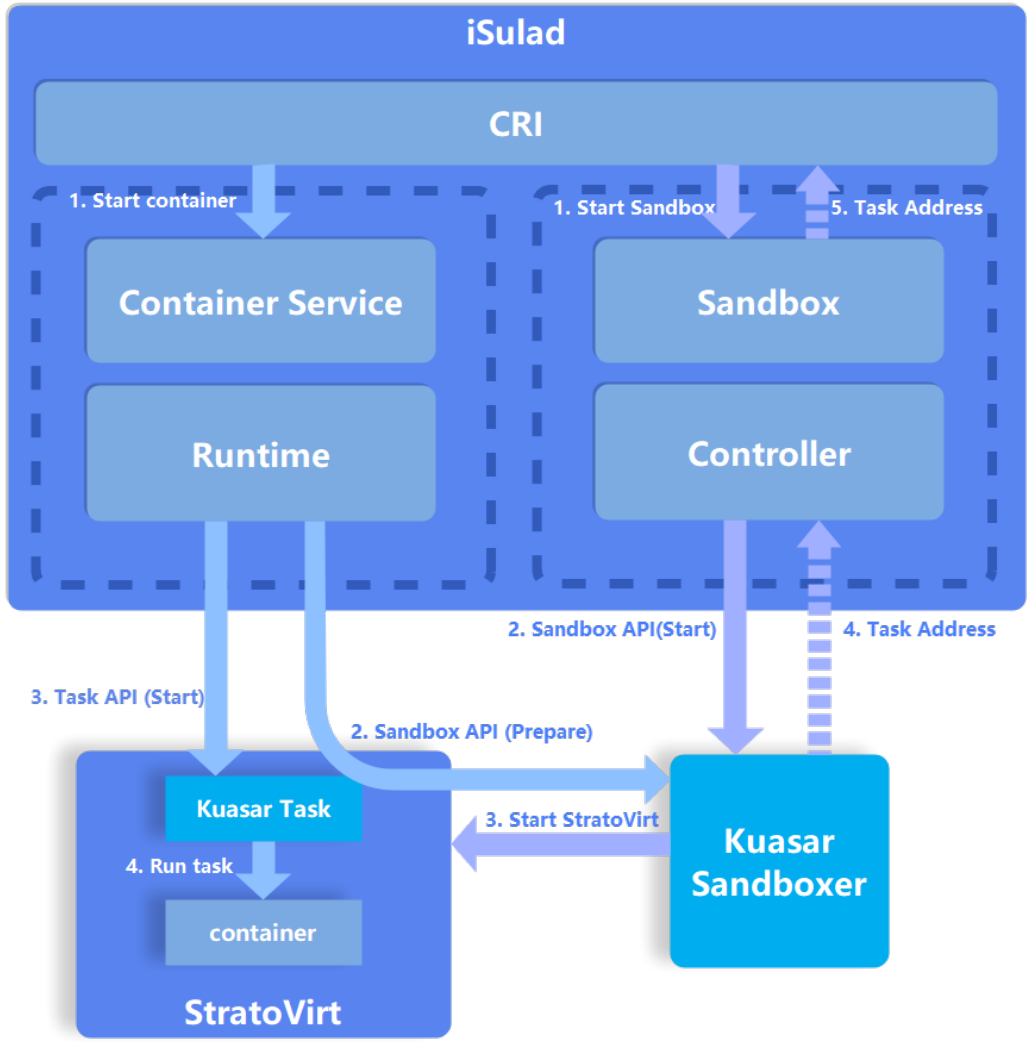
iSulad 对接 Kuasar

- iSulad 支持 Sandbox API 通过 Kuasar Sandboxer 进程直接管理沙箱生命周期
- iSulad 支持 Task API 直接管理容器生命周期，不需要 shim 进程协助管理

全栈轻量安全容器解决方案

容器启动流程

1. 根据CRI Container配置调用Container Service接口启动容器。
2. 调用Kuasar Sandboxer提供的资源管理接口，准备容器启动需要的资源。
3. 根据创建沙箱时返回的Task Address，调用Kuasar Task提供的Task API，启动容器。
4. Kuasar Task在虚机内启动容器任务。



沙箱启动流程

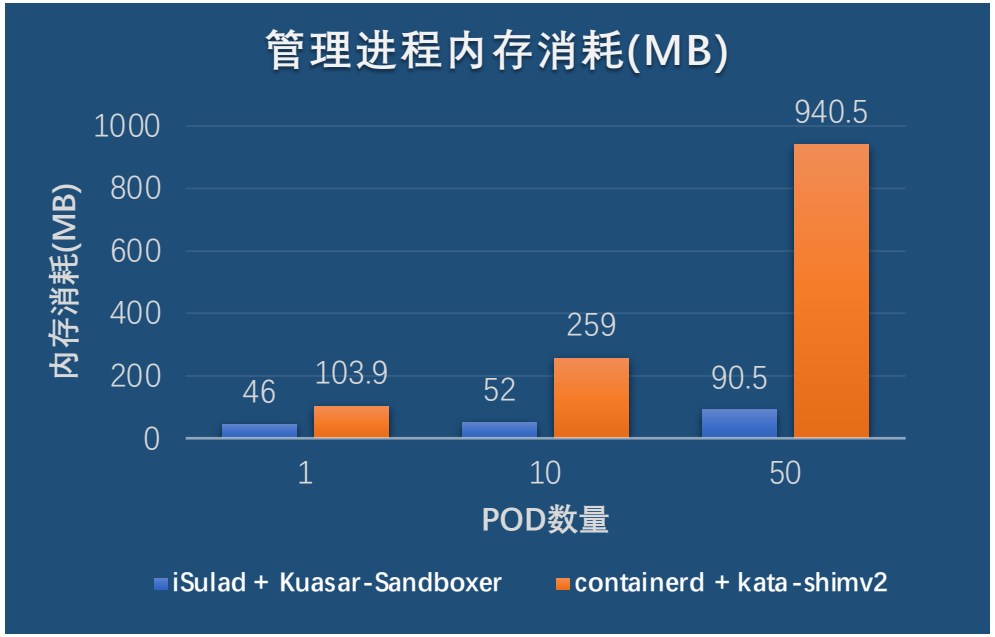
1. 根据CRI Pod配置创建Sandbox实例，并调用Start接口启动沙箱。
2. Sandbox实例通过Controller调用Kuasar Sandboxer进程提供的Sandbox API RPC。
3. Kuasar Sandboxer服务创建StratoVirt虚机，并返回虚机中Kuasar Task的地址，用于容器的管理。

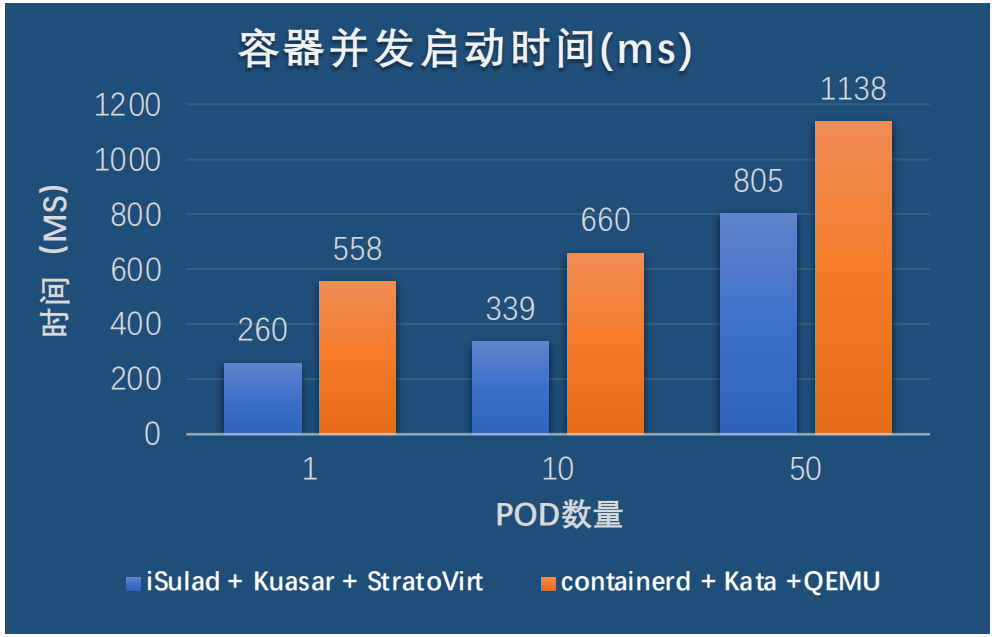
openEuler 23.09 一键部署基于 Kuasar 的极速轻量安全容器：详见【openEuler 文档 – 云原生 - 容器用户指南】

性能对比

- 管理面内存优化 **90%↑** 以上
- 容器并发启动时间优化 **30%↑** 以上

环境信息	
架构	x86
CPU核数	72
CPU频率	2.30GHz
内存	256G





THANKS