openEuler基于iSulad+Kuasar+StratoVirt的安全容器解决方案探索

姜鹏飞 华为

Content

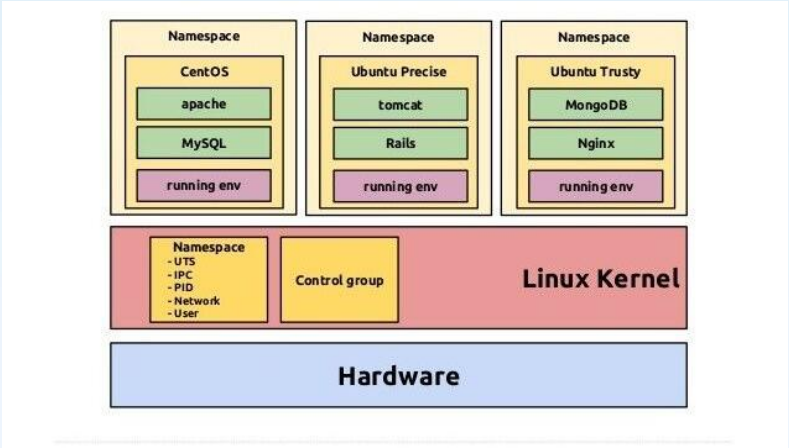
目录

Part 01

容器与沙箱介绍

云原生生态中的容器与沙箱

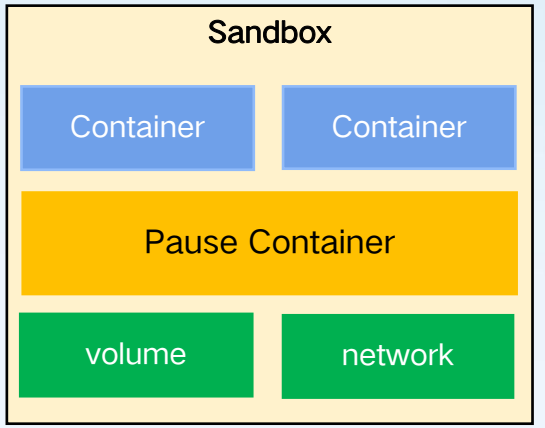
sandbox: 隔离运行程序的安全机制（VM/Seccomp/Wasm）
https://en.wikipedia.org/wiki/Sandbox_(computer_security)



container: 内核用来隔离用户进程的技术（lxc/runc）
https://en.wikipedia.org/wiki/OS-level_virtualization

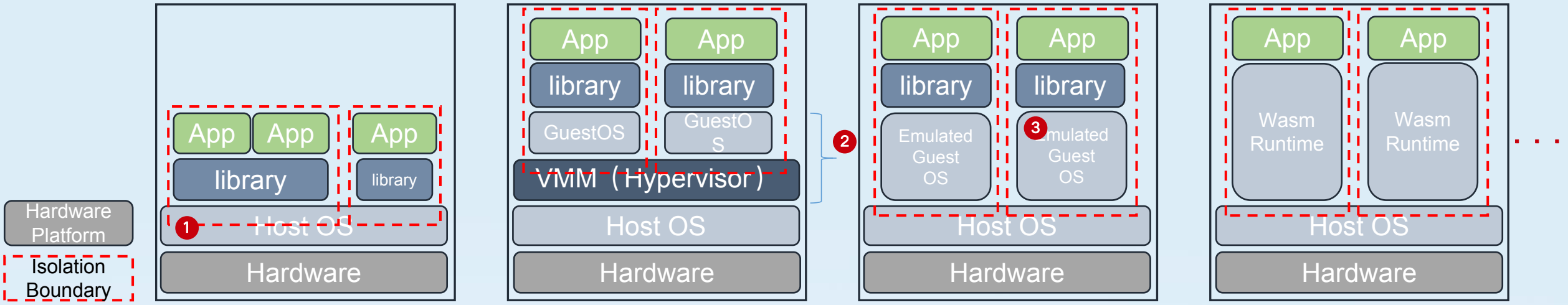
```
service RuntimeService {

    // Sandbox operations.

    rpc RunPodSandbox(RunPodSandboxRequest) returns (RunPodSandboxResponse) {}
    rpc StopPodSandbox(StopPodSandboxRequest) returns (StopPodSandboxResponse) {}
    rpc RemovePodSandbox(RemovePodSandboxRequest) returns (RemovePodSandboxResponse) {}
    rpc PodSandboxStatus(PodSandboxStatusRequest) returns (PodSandboxStatusResponse) {}
    rpc ListPodSandbox(ListPodSandboxRequest) returns (ListPodSandboxResponse) {}

    // Container operations.
    rpc CreateContainer(CreateContainerRequest) returns (CreateContainerResponse) {}
    rpc StartContainer(StartContainerRequest) returns (StartContainerResponse) {}
    rpc StopContainer(StopContainerRequest) returns (StopContainerResponse) {}
    rpc RemoveContainer(RemoveContainerRequest) returns (RemoveContainerResponse) {}
    rpc ListContainers(ListContainersRequest) returns (ListContainersResponse) {}
    rpc ContainerStatus(ContainerStatusRequest) returns (ContainerStatusResponse) {}

    ...
}
```

K8s CRI RuntimeService 接口定义



K8s Pod 概念模型

新型沙箱隔离技术不断出现，沙箱技术进入百家争鸣时代



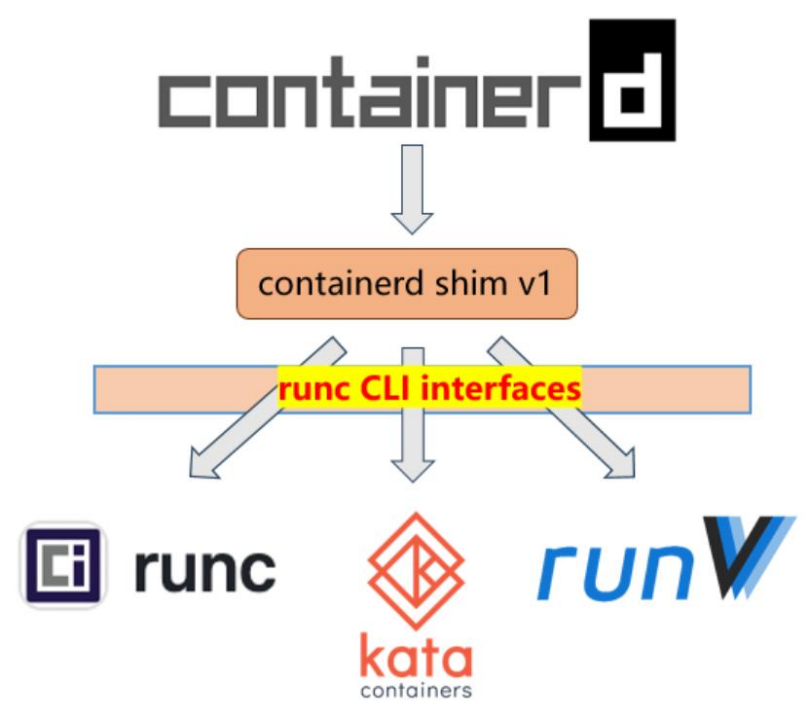
Runc 容器 | **microVM类** | **用户态内核类** | **WASM类**

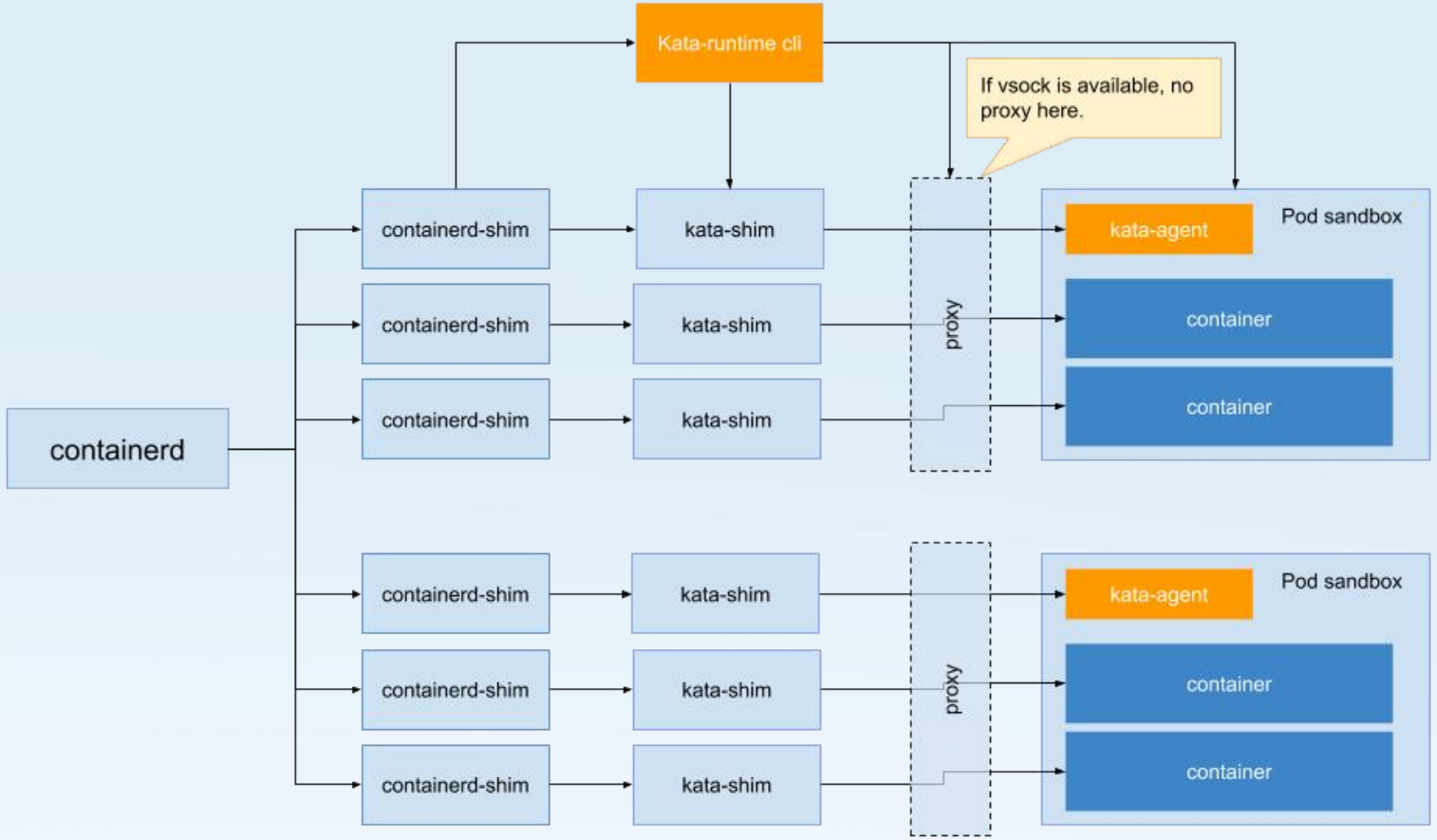
	Runc 容器	microVM类	用户态内核类	WASM类
技术原理	Linux容器+SELinux、AppArmor、Seccomp等安全机制	轻量级虚拟化技术	拦截用户所有系统调用，进程级虚拟化技术	底层虚拟机抽象，Runtime虚拟化技术
特点	• **标准通用：**共享内核，方便易用 • **资源效率：**原生容器，高性能、低开销	• **安全隔离：**具备完整的OS 和内核，提供虚机级别的隔离性能 • **标准通用：**原生linux支持，应用兼容性优	• **资源效率：**轻量级，启动性能接近Runc • **安全隔离：**提供独立内核，隔离性好	• **资源效率：**可以轻松实现毫秒级冷启动时间和极低的资源消耗
缺点	• **安全隔离问题：**无法有效防范内核安全问题；性能隔离不够健壮	• **资源效率问题：容器的VMM和GuestOS**额外内存开销较大，启动速度相对较慢	• **标准通用问题：**非标准linux，存在应用兼容的限制；不支持设备热插拔	• **标准通用问题：**缺乏标准化的网络访问能力，仅能做一些计算类任务 • **安全隔离：**部分WASM虚机无法对内存和CPU资源精确限制，无IO资源隔离能力

Part 02

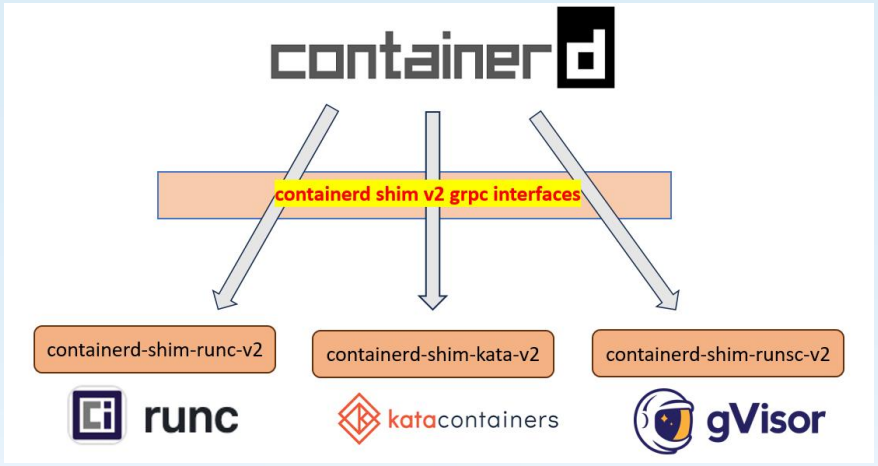
安全容器现状与挑战

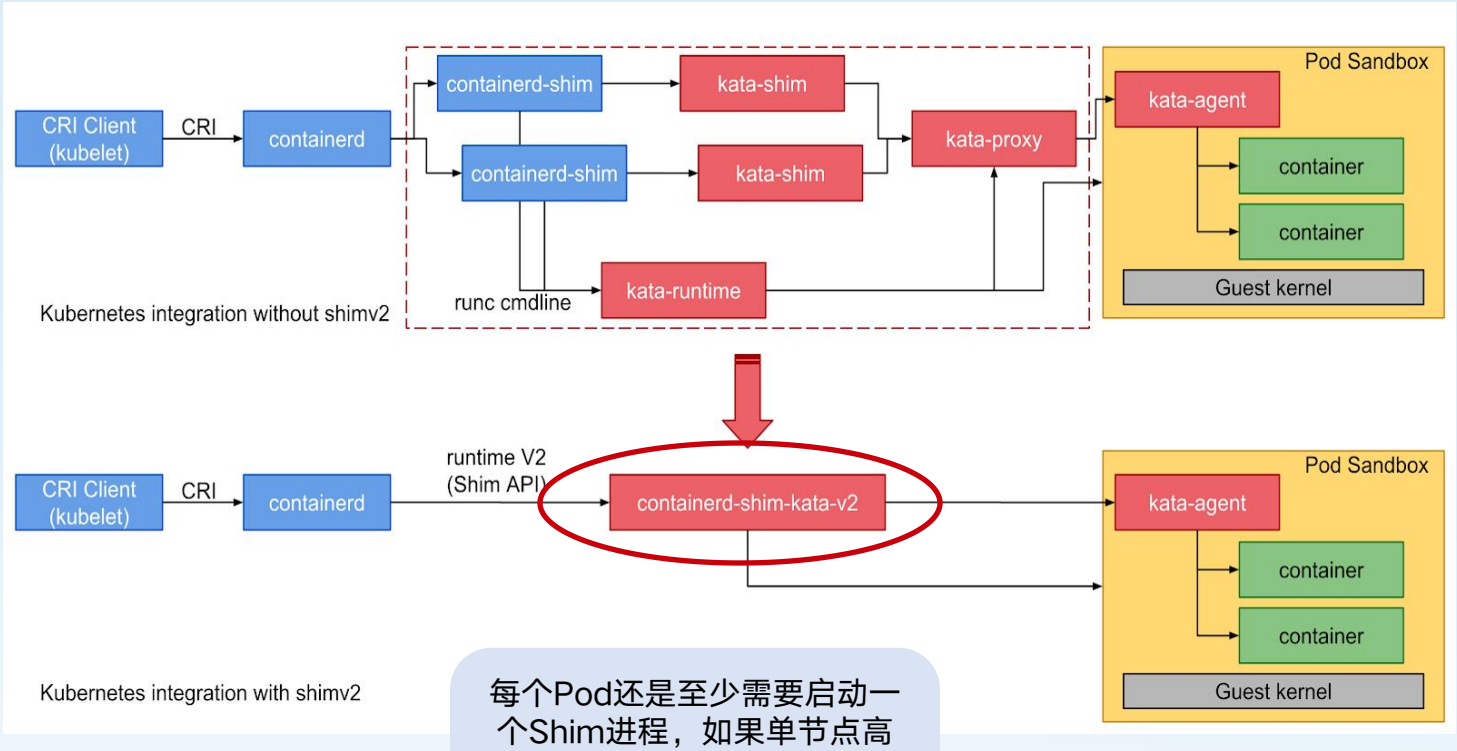
业界现状：shimv1 API导致shim进程数量爆炸





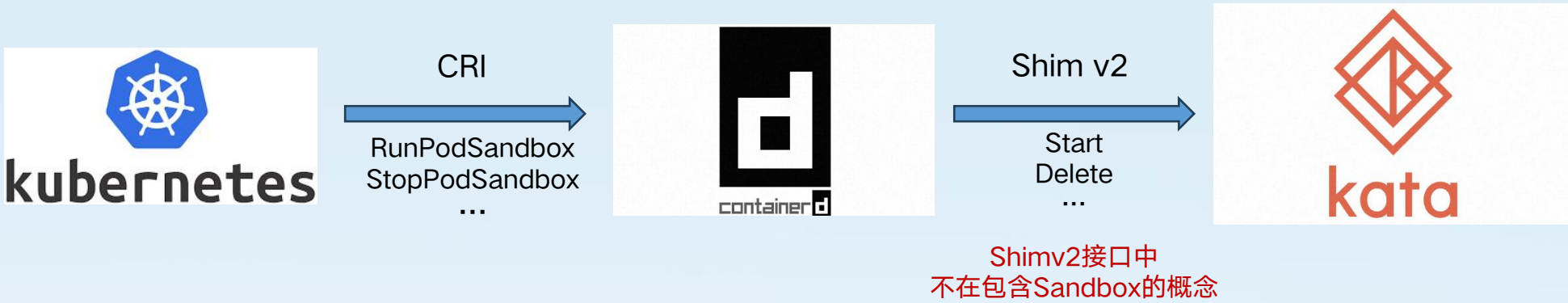
业界现状：shimv2 API优化了shim进程数量，但仍需要one-shim/per-pod





每个Pod还是至少需要启动一个Shim进程，如果单节点高密部署1000个容器，就需要1000个Shim进程，约10GB内存资源消耗

挑战：能否打破shim进程与Pod之间1:1的关系，进一步减少shim进程资源消耗？



关键洞察：Sandbox在Containerd中不是一等公民，缺失了沙箱定义，导致Shimv2 API中Sandbox操作与Container操作混淆交叉，需要通过Shim进程根据元数据信息进行区分，增加了实现成本

Part 03

Kuasar 安全容器解决方案介绍

云原生多沙箱容器运行时 - Kuasar

痛 点

云原生场景需求多

云原生场景不同，需求不同，孵化出的沙箱容器运行时不同

运维操作复杂度高

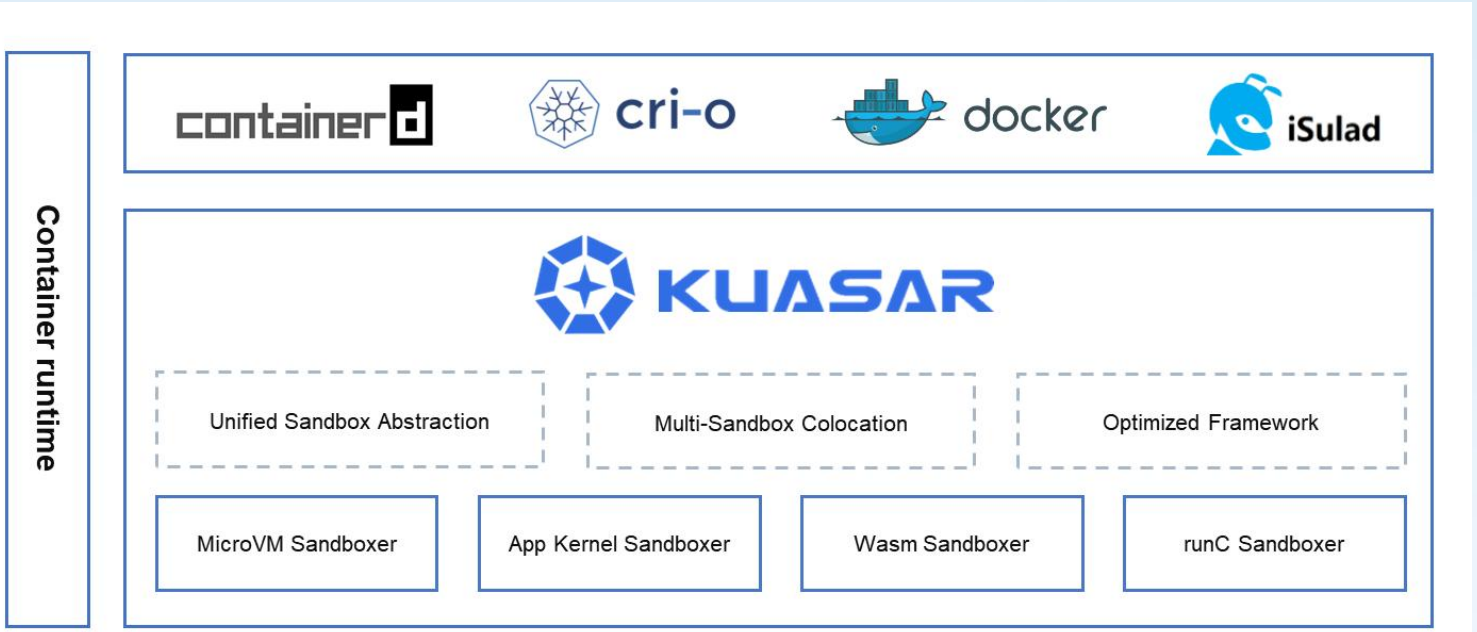
太多的沙箱容器运行时需要运维，复杂度极高

平滑迁移路径缺失

沙箱迁移成本高，难以拥抱新沙箱的出现

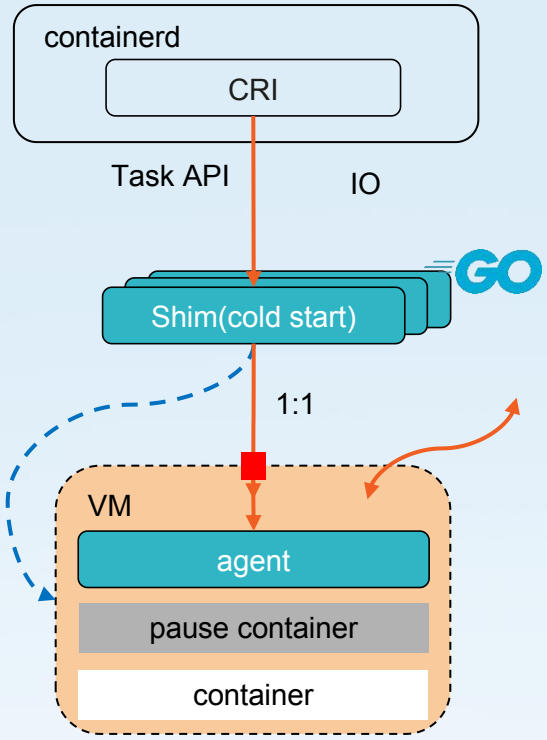
KUASAR

统一沙箱定义 | 多沙箱混部 | 极致框架优化

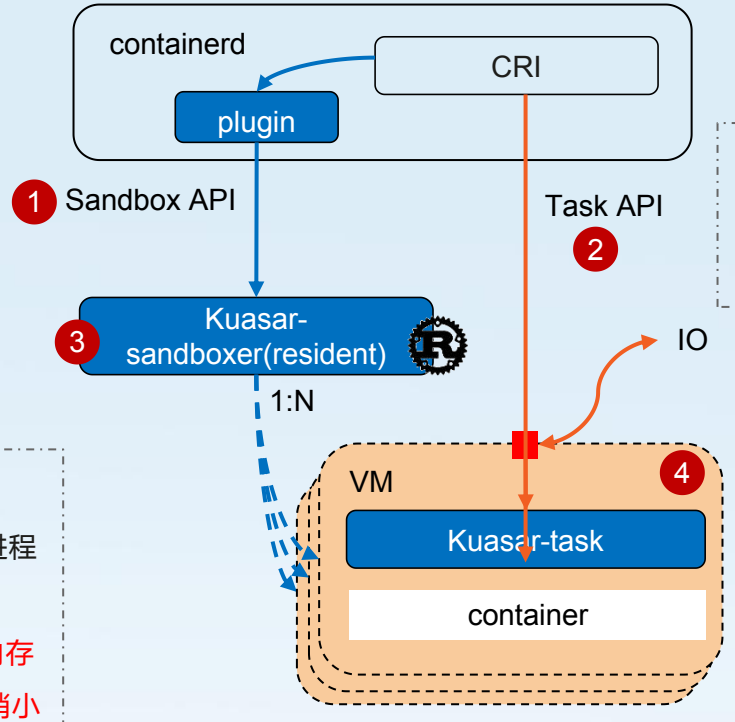


Kuasar 架构优化

当前 Shim v2 API 模型



Kuasar 的 Sandbox API 模型



sandbox 管理逻辑清晰

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

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 管理模型减少了进程数量，节省大量内存

c. Rust 程序内存安全，相比 Golang 内存开销小

简化 container API 调用链

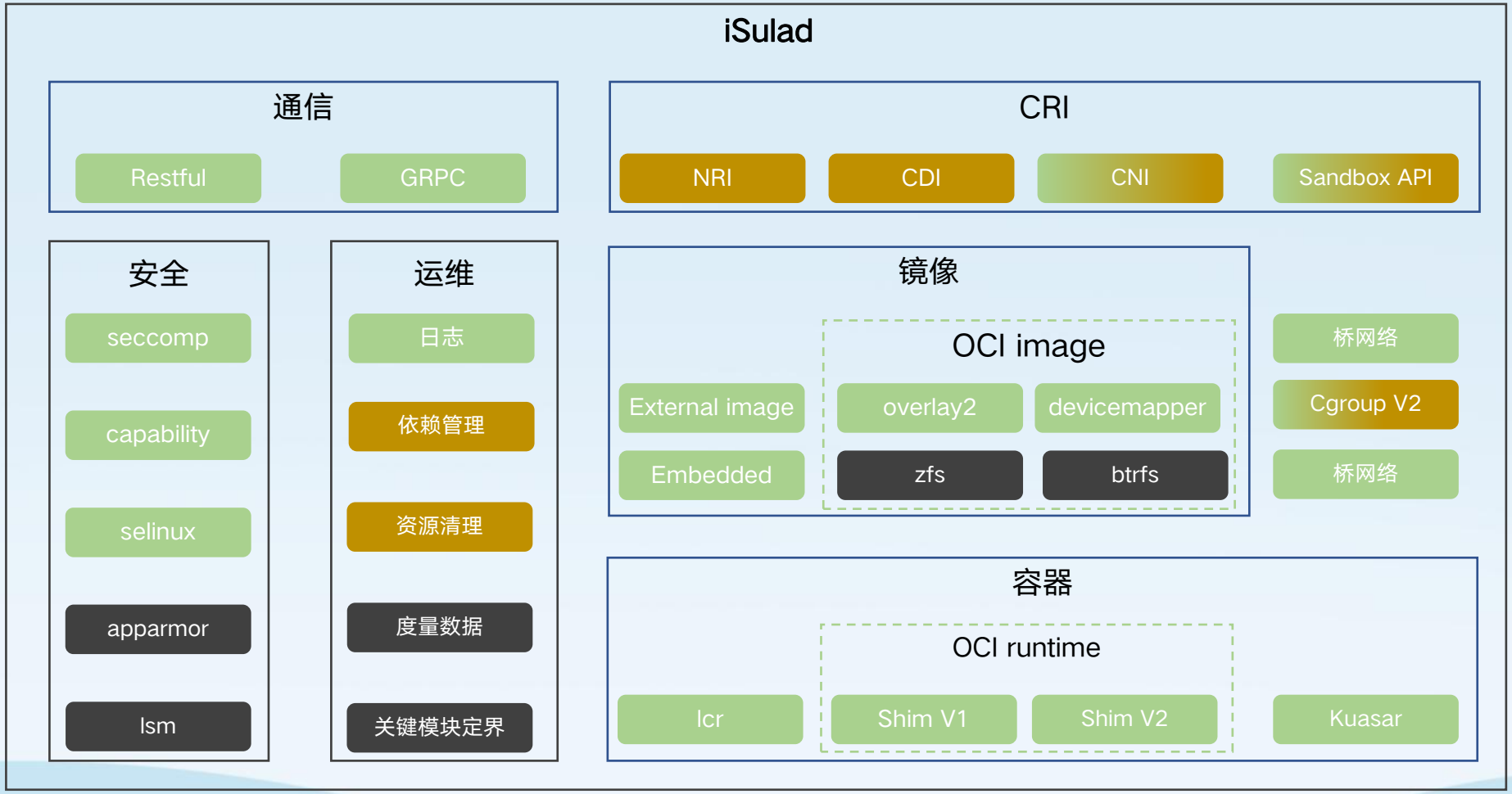
取消 Task API 到 Shim v2 API 的转化，链路简化， Pod 启动速度加快

pause 容器消失

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

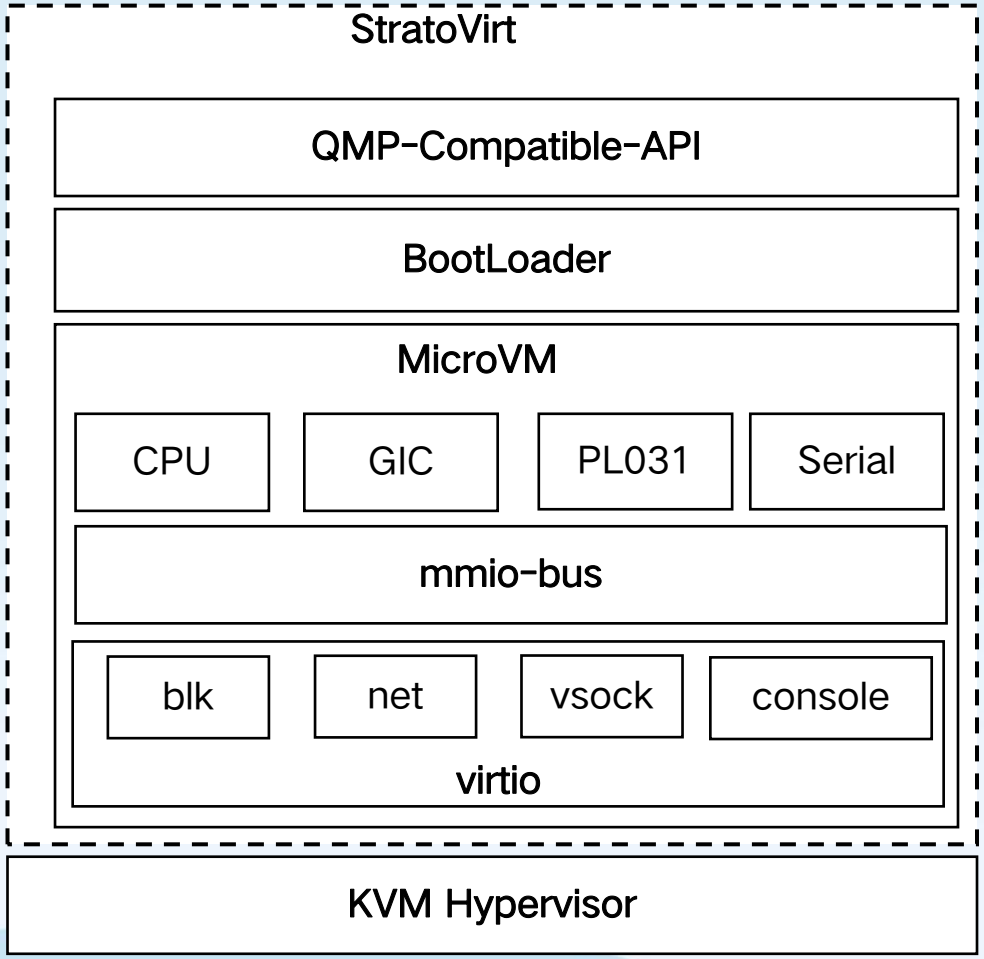
轻量级容器引擎 - iSulad

iSulad是面向端、边、云全场景的轻量级容器引擎，兼容云原生社区生态，具有**轻、灵、巧、快**的特点



轻量级虚拟化引擎 - StratoVirt

StratoVirt是面向云数据中心的企业级虚拟化平台，实现一套架构对**虚拟机、容器、Serverless**三种场景的统一支持



强安全性

采用Rust语言，支持seccomp，减小系统攻击面，实现多租户安全隔离

轻量低噪

采用极简设备模型时，启动时间<50ms，内存底噪<4M，支持Serverless负载

极速伸缩

毫秒级设备扩缩能力，为轻量化负载提供灵活的资源伸缩能力

软硬协同

同时支持x86的VT和鲲鹏的Kunpeng-V，实现多体系硬件加速

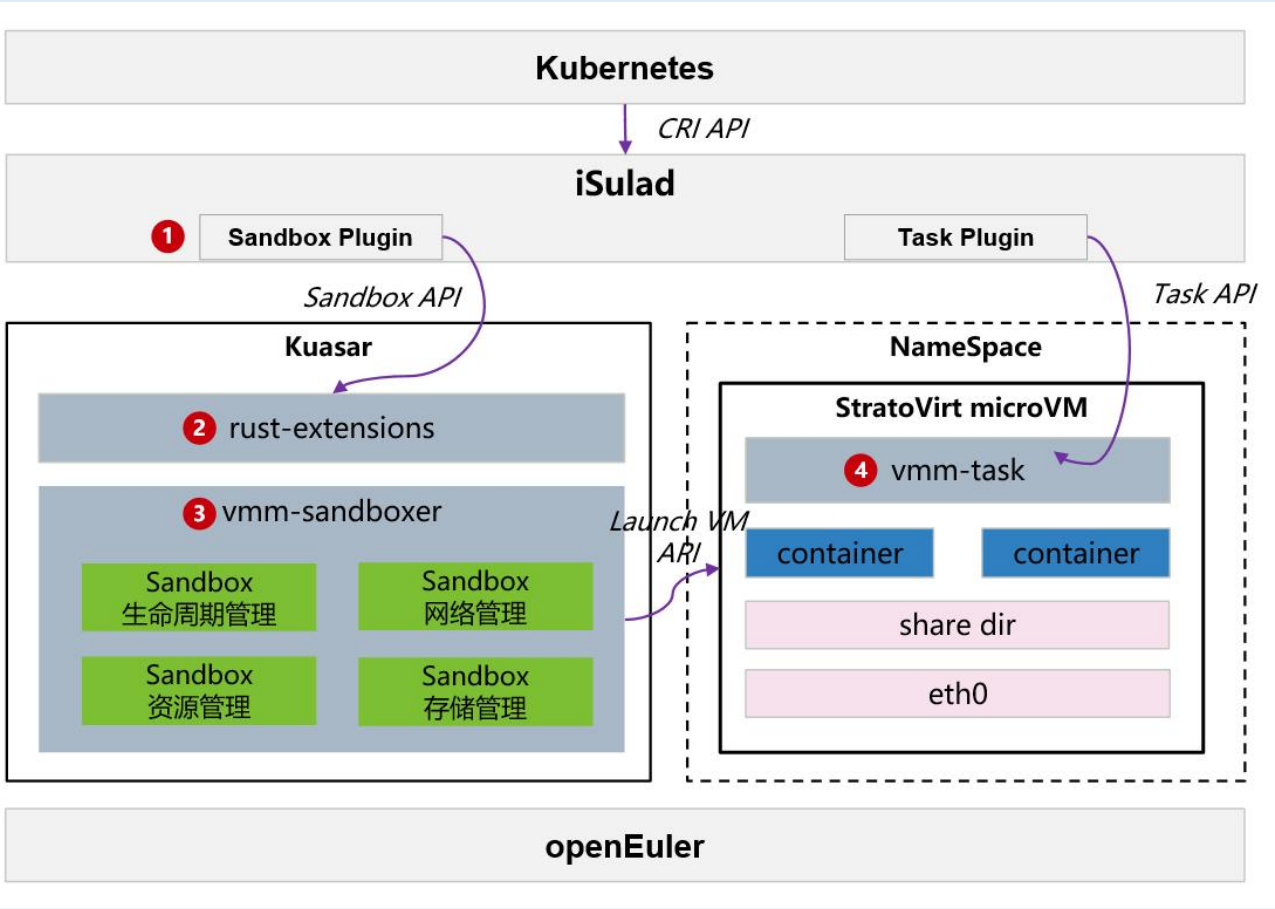
高扩展性

设备模型可扩展，支持PCI等复杂设备规范，实现标准虚拟机

异构增强

除支持常用的硬件SR-IOV直通方案，结合昇腾软件定义能力，实现更灵活异构算力分配

iSulad + Kuasar + StratoVirt 安全容器沙箱解决方案



iSulad+Kuasar+StratoVirt 安全容器沙箱架构图

1. iSulad Sandbox Plugin：容器引擎沙箱管理插件

iSulad容器引擎中新增sandbox沙箱对象管理模块，支持通过sandbox API来管理sandbox对象的生命周期管理。

2. rust-extensions：沙箱统一管理模块

rust-extensions是Kuasar容器运行时抽象出来统一对接Sandbox API的公共库模块，根据容器引擎层下发的Sandbox API请求调用相应的sandbox类型的sandboxer，完成相应具体sandbox对象的创建、删除等生命周期管理操作。

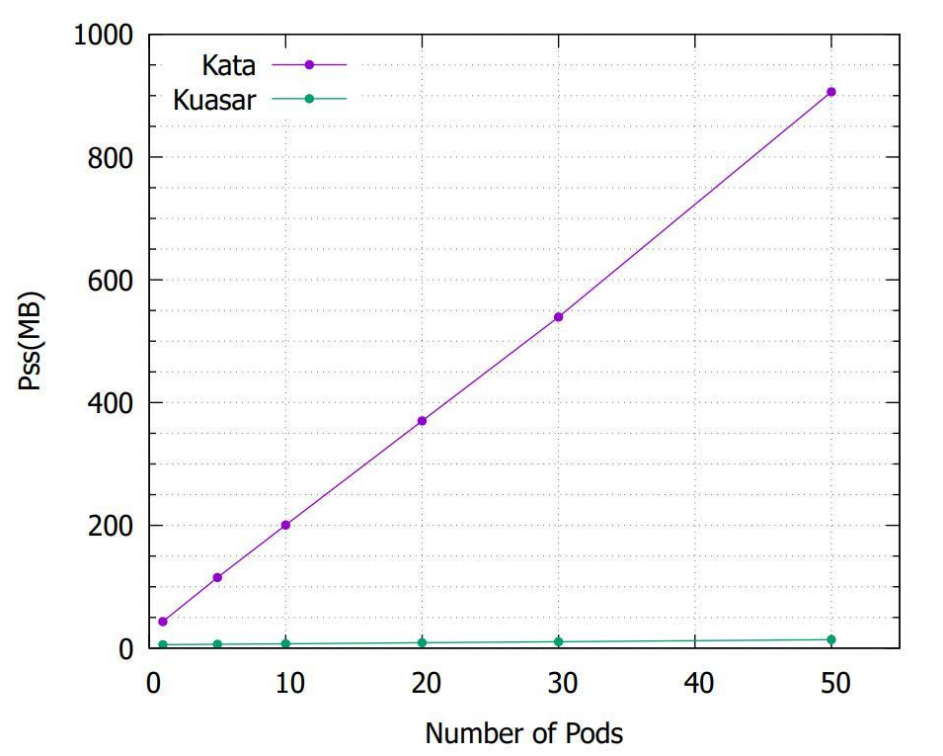
3. vmm-sandboxer：轻量级虚拟机类型沙箱管理模块

提供了vmm类型sandbox生命周期管理、sandbox资源管理、sandbox网络管理以及sandbox存储管理的功能，通过VM API接口实现对虚拟机对象的生命周期管理，如创建/停止/删除虚拟机对象。

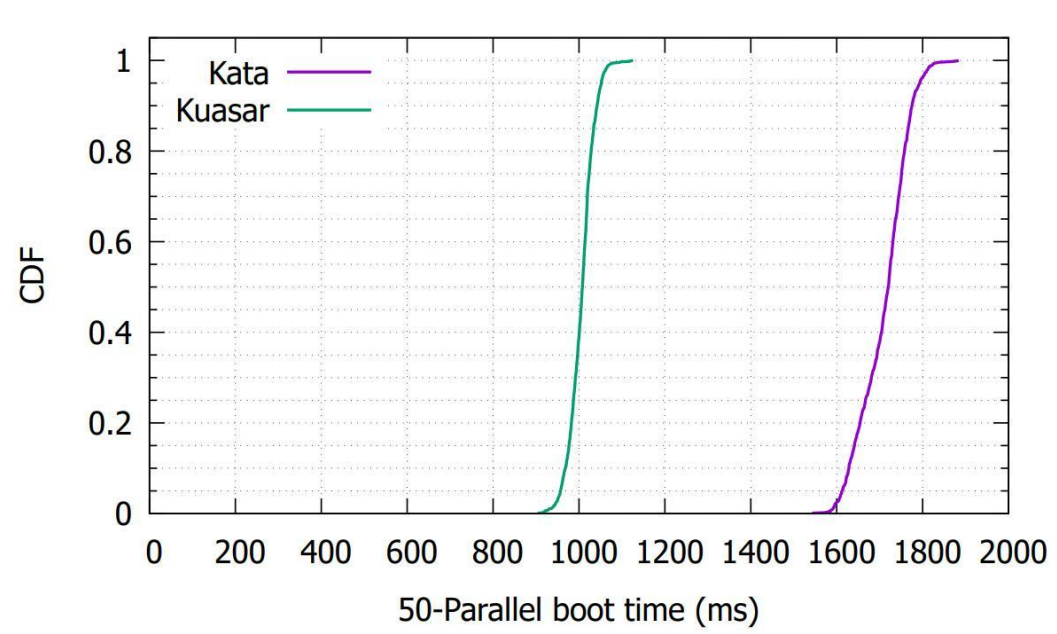
4. vmm-task：沙箱任务管理模块

vmm-task 模块是虚拟机沙箱中1号进程，负责响应从iSulad容器引擎下发的Task API请求，对业务容器的生命周期和资源分配进行管理。

性能测试：管理面内存占用降低99%，并发启动速度提升2x



kuasar 管理面内存消耗降低99%



kuasar 并发启动速度提升2x

Part 04

未来展望

未来展望

2023 H1	2023 H2	2024	2025
● MicroVM沙箱 ✓ StratoVirt ✓ Cloud Hypervisor ✓ QEMU ● App Kernel沙箱 ✓ Quark ● Wasm 沙箱 ✓ WasmEdge ● 支持对接 iSulad ● 支持对接 containerd v1.7.0	● Wasm 沙箱 ✓ Wasmtime ● 支持 aarch64 架构 ● Guest kernel 裁剪优化	● 支持 Runc 普通容器 ● 支持对接 containerd v2.0 ● MicroVM沙箱 ✓ Firecracker ● App Kernel沙箱 ✓ gVisor ● 推出命令行工具	● 支持机密容器 ● 镜像下载加速 ● eBPF 可观测

欢迎大家关注openEuler社区iSulad/Kuasar/StratoVirt项目

https://gitee.com/openeuler/iSulad

https://gitee.com/src-openeuler/kuasar
https://github.com/kuasar-io/kuasar

StratoVirt

https://gitee.com/openeuler/stratovirt

Thanks.