openEuler 24.03 LTS 技术白皮书

1. 概述

OpenAtom openEuler（简称“openEuler”） 社区是一个面向数字基础设施操作系统的开源社区。由开放原子开源基金会（以下简称“基金会”）孵化及运营。

openEuler 是一个面向数字基础设施的操作系统，支持服务器、云计算、边缘计算、嵌入式等应用场景，支持多样性计算，致力于提供安全、稳定、易用的操作系统。通过为应用提供确定性保障能力，支持 OT 领域应用及 OT 与 ICT 的融合。

openEuler 社区通过开放的社区形式与全球的开发者共同构建一个开放、多元和架构包容的软件生态体系，孵化支持多种处理器架构、覆盖数字基础设施全场景，推动企业数字基础设施软硬件、应用生态繁荣发展。

2019 年 12 月 31 日，面向多样性计算的操作系统开源社区 openEuler 正式成立。

2020 年 3 月 30 日， openEuler 20.03 LTS （Long Term Support，简写为 LTS，中文为长生命周期支持）版本正式发布，为 Linux 世界带来一个全新的具备独立技术演进能力的 Linux 发行版。

2020 年 9 月 30 日，首个 openEuler 20.09 创新版发布，该版本是 openEuler 社区中的多个企业、团队、独立开发者协同开发的成果，在 openEuler 社区的发展进程中具有里程碑式的意义，也是中国开源历史上的标志性事件。

2021 年 3 月 31 日，发布 openEuler 21.03 内核创新版，该版本将内核升级到 5.10，并在内核方向实现内核热升级、内存分级扩展等多个创新特性，加速提升多核性能，构筑千核运算能力。

2021 年 9 月 30 日，全新 openEuler 21.09 创新版如期而至，这是 openEuler 全新发布后的第一个社区版本，实现了全场景支持。增强服务器和云计算的特性，发布面向云原生的业务混部 CPU 调度算法、容器化操作系统 KubeOS 等关键技术；同时发布边缘和嵌入式版本。

2022 年 3 月 30 日，基于统一的 5.10 内核，发布面向服务器、云计算、边缘计算、嵌入式的全场景 openEuler 22.03 LTS 版本，聚焦算力释放，持续提升资源利用率，打造全场景协同的数字基础设施操作系统。

2022 年 9 月 30 日，发布 openEuler 22.09 创新版本，持续补齐全场景的支持。

2022 年 12 月 30 日，发布 openEuler 22.03 LTS SP1 版本，打造最佳迁移工具实现业务无感迁移，性

能持续领先。

2023 年 3 月 30 日，发布 openEuler 23.03 内核创新版本，采用 Linux Kernel 6.1 内核，为未来 openEuler 长生命周期版本采用 6.x 内核提前进行技术探索，方便开发者进行硬件适配、基础技术创新及上层应用创新。

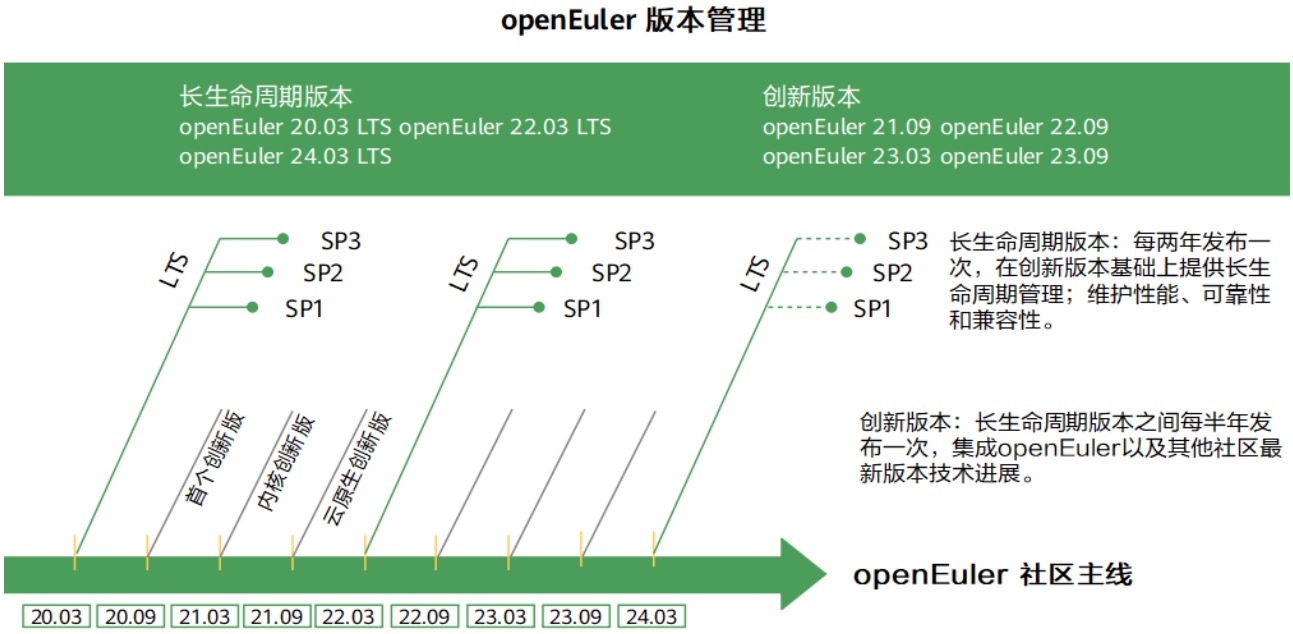
2023 年 6 月 30 日，发布 openEuler 22.03 LTS SP2 版本，场景化竞争力特性增强，性能持续提升。

2023 年 9 月 30 日，发布 openEuler 23.09 创新版本，是基于 6.4 内核的创新版本（参见版本生命周期），提供更多新特性和功能，给开发者和用户带来全新的体验，服务更多的领域和更多的用户。

2023 年 11 月 30 日，发布 openEuler 20.03 LTS SP4 版本，其作为 20.03 LTS 版本的增强扩展版本，面向服务器、云原生、边缘计算场景，提供更多新特性和功能增强。

2023 年 12 月 30 日，发布 openEuler 22.03 LTS SP3 版本，是 22.03 LTS 版本增强扩展版本，面向服务器、云原生、边缘计算和嵌入式场景，持续提供更多新特性和功能扩展，给开发者和用户带来全新的体验，服务更多的领域和更多的用户。

2024 年 5 月 30 日，发布 openEuler 24.03 LTS，基于 6.6 内核的长周期 LTS 版本（参见版本生命周期），面向服务器、云、边缘计算、AI 和嵌入式场景，提供更多新特性和功能，给开发者和用户带来全新的体验，服务更多的领域和更多的用户。



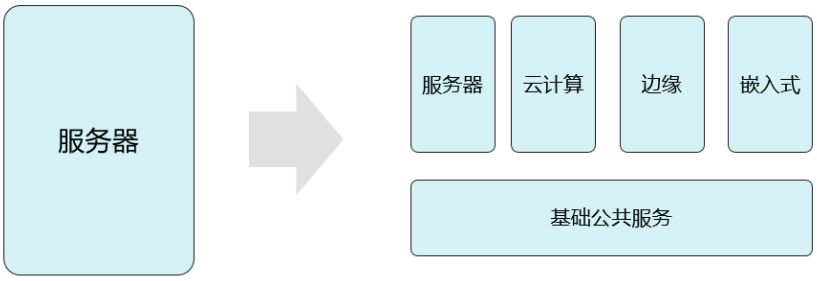
openEuler 作为一个操作系统发行版平台，每两年推出一个 LTS 版本。该版本为企业级用户提供一个安全稳定可靠的操作系统。

openEuler 也是一个技术孵化器。通过每半年发布一个创新版，快速集成 openEuler 以及其他社区的最新技术成果，将社区验证成熟的特性逐步回合到发行版中。这些新特性以单个开源项目的方式存在于社

区，方便开发者获得源代码，也方便其他开源社区使用。

社区中的最新技术成果持续合入社区发行版，社区发行版通过用户反馈反哺技术，激发社区创新活力，从而不断孵化新技术。发行版平台和技术孵化器互相促进、互相推动、牵引版本持续演进。

openEuler 覆盖全场景的创新平台



openEuler 已支持 x86、Arm、SW64、RISC-V、LoongArch 多处理器架构，逐步扩展 PowerPC 等更多芯片架构支持，持续完善多样性算力生态体验。

openEuler 社区面向场景化的 SIG 不断组建，推动 openEuler 应用边界从最初的服务器场景，逐步拓展到云计算、边缘计算、嵌入式等更多场景。openEuler 正成为覆盖数字基础设施全场景的操作系统，新增发布面向边缘计算的版本 openEuler Edge、面向嵌入式的版本 openEuler Embedded。

openEuler 希望与广大生态伙伴、用户、开发者一起，通过联合创新、社区共建，不断增强场景化能力，最终实现统一操作系统支持多设备，应用一次开发覆盖全场景。

openEuler 开放透明的开源软件供应链管理

开源操作系统的构建过程，也是供应链聚合优化的过程。拥有可靠开源软件供应链，是大规模商用操作系统的基础。 openEuler 从用户场景出发，回溯梳理相应的软件依赖关系，理清所有软件包的上游社区地址、源码和上游对应验证。完成构建验证、分发、实现生命周期管理。开源软件的构建、运行依赖关系、上游社区，三者之前形成闭环且完整透明的软件供应链管理。

2. 平台架构

系统框架

openEuler 是覆盖全场景的创新平台，在引领内核创新，夯实云化基座的基础上，面向计算架构互联总线、存储介质发展新趋势，创新分布式、实时加速引擎和基础服务，结合边

缘、嵌入式领域竞争力探索，打造全场景协同的面向数字基础设施的开源操作系统。

openEuler 24.03 LTS 发布面向服务器、云原生、边缘和嵌入式场景的全场景操作系统版本，统一基于 Linux Kernel 6.6 构建，对外接口遵循 POSIX 标准，具备天然协同基础。同时 openEuler 24.03 LTS 版本集成分布式软总线、KubeEdge+边云协同框架等能力，进一步提升数字基础设施协同能力，构建万物互联的基础。

面向未来，社区将持续创新、社区共建、繁荣生态，夯实数字基座。

夯实云化基座

- 容器操作系统 KubeOS：云原生场景，实现 OS 容器化部署、运维，提供与业务容器一致的基于 K8S 的管理体验。
- 安全容器方案：iSulad+shimv2+StratoVirt 安全容器方案，相比传统 Docker+QEMU 方案，底噪和启动时间优化 40%。
- 双平面部署工具 eggo：Arm/x86 双平面混合集群 OS 高效一键式安装，百节点部署时间 <15min。

新场景

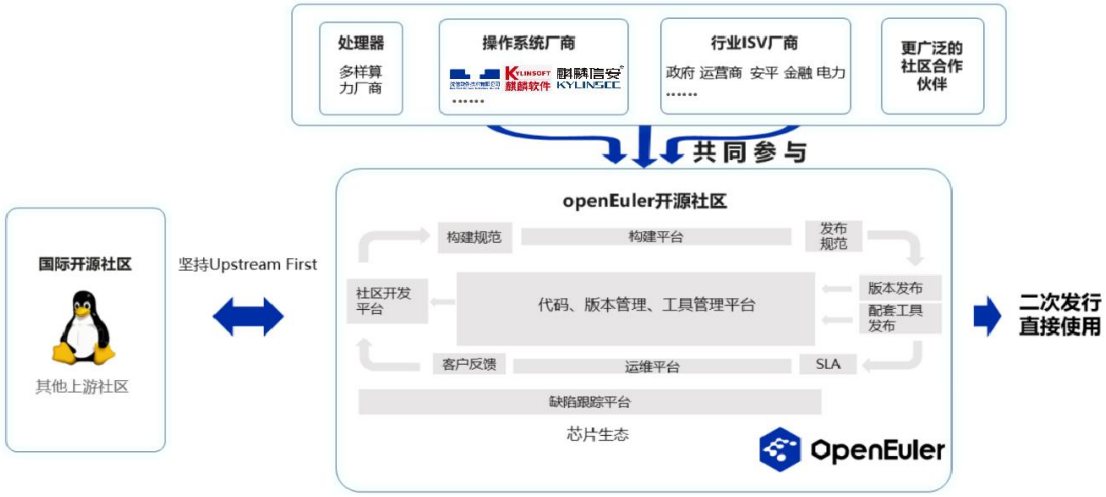
- 边缘计算：发布面向边缘计算场景的版本 openEuler 24.03 LTS Edge，支持 KubeEdge+边云协同框架，具备边云应用统一管理和发放等基础能力。
- 嵌入式： 发布面向嵌入式领域的版本 openEuler 24.03 LTS Embedded，镜像大小 < 5M，启动时间 < 5s。

繁荣社区生态

- 友好桌面环境：UKUI、DDE 、Xfce、Kiran-desktop、GNOME 桌面环境，丰富社区桌面环境生态。
- openEulerDevKit：支持操作系统迁移、兼容性评估、简化安全配置 secPaver 等更多开发工具。

平台框架

openEuler 社区与上下游生态建立连接，构建多样性的社区合作伙伴和协作模式，共同推进版本演进。



硬件支持

openEuler 社区当前已与多个设备厂商建立丰富的南向生态，比如 Intel、AMD 等主流芯片厂商的加入和参与，openEuler 全版本支持 x86、Arm、申威、龙芯、RISC-V 五种架构，并支持多款 CPU 芯片，包括龙芯 3 号 、兆芯开先/开胜系列、Intel IceLake/ Sapphire Rapids、AMD EPYC Milan /Genoa 等芯片系列，支持多个硬件厂商发布的多款整机型号、板卡型号，支持网卡、RAID、FC、GPU&AI、DPU、SSD、安全卡七种类型的板卡，具备良好的兼容性。

支持的 CPU 架构如下：

硬件类型	x86	Arm	LoongArch	SW64	RISC-V
CPU	Intel、AMD、兆芯、海光	鲲鹏、飞腾	龙芯	申威	Sophgo、T-Head 等

支持的整机如下：

硬件类型	x86	Arm
整机	•**Intel**：超聚变、超微 •**AMD**：超微、新华三 •**海光**：曙光/中科可控	•**鲲鹏**：泰山 •**飞腾**：青松、宝德、新华三
	•**兆芯**：兆芯	

支持的板卡类型如下：

硬件类型	x86	Arm
网卡	华为、Mellanox、Intel、Broadcom、云芯智联、Netswift、云脉、沐创	华为、Mellanox、Intel、Broadcom、云芯智联、Netswift、云脉、沐创

Raid	Avago、云芯智联、PMC、华为	Avago、云芯智联、PMC、华为
FC	Marvell、Qlogic、Emulex	Marvell、Qlogic、Emulex
GPU&AI	Nvidia	Nvidia
SSD	华为	华为

全版本支持的硬件型号可在兼容性网站查询：https://www.openeuler.org/zh/compatibility/。

3. 运行环境

服务器

若需要在物理机环境上安装 openEuler 操作系统，则物理机硬件需要满足以下兼容性和最小硬件要求。

硬件兼容支持请查看 openEuler 兼容性列表：https://openeuler.org/zh/compatibility/。

部件名称	最小硬件要求
架构	Arm64、x86_64
内存	为了获得更好的体验，建议不小于 4GB
硬盘	为了获得更好的体验，建议不小于 20GB

虚拟机

openEuler 安装时，应注意虚拟机的兼容性问题，当前已测试可以兼容的虚拟机及组件如下所示。

1.以 openEuler 24.03 LTS 为 HostOS，组件版本如下：

- libvirt-9.10.0-9.oe2403
- libvirt-client-9.10.0-9.oe2403
- libvirt-daemon-9.10.0-9.oe2403
- qemu-8.2.0-12.oe2403
- qemu-img-8.2.0-12.oe2403

2.**兼容的虚拟机列表如下：**

HostOS	GuestOS(虚拟机)	架构

openEuler 24.03 LTS	Centos 6	x86_64
openEuler 24.03 LTS	Centos 7	aarch64
openEuler 24.03 LTS	Centos 7	x86_64
openEuler 24.03 LTS	Centos 8	aarch64
openEuler 24.03 LTS	Centos 8	x86_64
openEuler 24.03 LTS	Windows Server 2016	aarch64
openEuler 24.03 LTS	Windows Server 2016	x86_64
openEuler 24.03 LTS	Windows Server 2019	aarch64
openEuler 24.03 LTS	Windows Server 2019	x86_64

部件名称	最小虚拟化空间要求
架构	Arm64、x86_64
CPU	2 个 CPU
内存	为了获得更好的体验，建议不小于 4GB
硬盘	为了获得更好的体验，建议不小于 20GB

边缘设备

若需要在边缘设备环境上安装 openEuler 操作系统，则边缘设备硬件需要满足以下兼容性和最小硬件要求。

部件名称	最小硬件要求
架构	Arm64、x86_64
内存	为了获得更好的体验，建议不小于 4GB
硬盘	为了获得更好的体验，建议不小于 20GB

嵌入式

若需要在嵌入式环境上安装 openEuler 操作系统，则嵌入式硬件需要满足以下兼容性和最小硬件要求。

部件名称	最小硬件要求
架构	Arm64、Arm32
内存	为了获得更好的体验，建议不小于 512MB
硬盘	为了获得更好的体验，建议不小于 256MB

4. 场景创新

AI

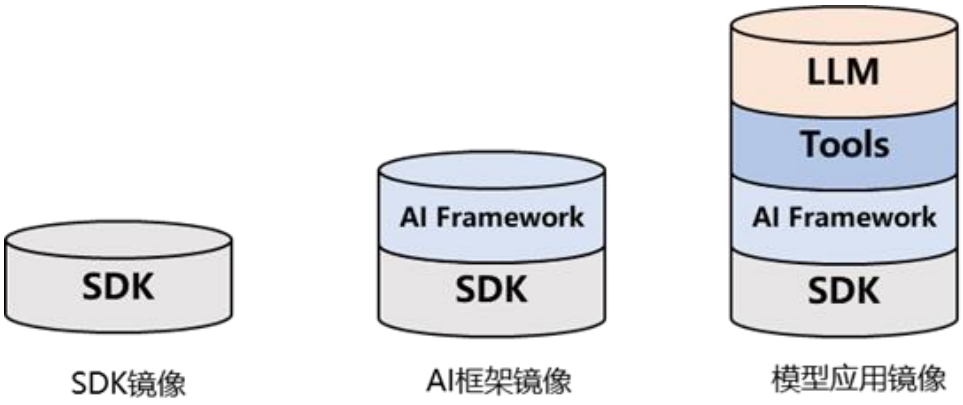
智能时代，操作系统需要面向 AI 不断演进。一方面，在操作系统开发、部署、运维全流程以 AI 加持，让操作系统更智能；另一方面，openEuler 已支持 Arm，x86，RISC-V 等全部主流通用计算架构，在智能时代，openEuler 也率先支持 NVIDIA、昇腾等主流 AI 处理器，成为使能多样性算力的首选。

OS for AI

openEuler 兼容 NVIDIA、Ascend 等主流算力平台的软件栈，为用户提供高效的开发运行环境。通过将不同 AI 算力平台的软件栈进行容器化封装，即可简化用户部署过程，提供开箱即用的体验。同时，openEuler 也提供丰富的 AI 框架，方便大家快速在 openEuler 上使用 AI 能力。

功能描述

1. openEuler 已兼容 CANN、CUDA 等硬件 SDK，以及 TensorFlow、PyTorch 等相应的 AI 框架软件，支持 AI 应用在 openEuler 上高效开发与运行。
2. openEuler AI 软件栈容器化封装优化环境部署过程，并面向不同场景提供以下三类容器镜像。



- SDK 镜像：以 openEuler 为基础镜像，安装相应硬件平台的 SDK，如 Ascend 平台的 CANN 或 NVIDIA 的 CUDA 软件。
- AI 框架镜像：以 SDK 镜像为基础，安装 AI 框架软件，如 PyTorch 或 TensorFlow。
- 模型应用镜像：在 AI 框架镜像的基础上，包含完整的工具链和模型应用。

相关使用方式请参考 openEuler AI 容器镜像用户指南。

应用场景

openEuler 使能 AI，向用户提供更多 OS 选择。基于 openEuler 的 AI 容器镜像可以解决开发运行环境部署门槛高的问题，用户根据自身需求选择对应的容器镜像即可一键部署，三类容器镜像的应用场景如下。

- SDK 镜像：提供对应硬件的计算加速工具包和开发环境，用户可进行 Ascend CANN 或 NVIDIA CUDA 等应用的开发和调试。同时，可在该类容器中运行高性能计算任务，例如大规模数据处理、并行计算等。
- AI 框架镜像：用户可直接在该类容器中进行 AI 模型开发、训练及推理等任务。
- 模型应用镜像：已预置完整的 AI 软件栈和特定的模型，用户可根据自身需求选择相应的模型应用镜像来开展模型推理或微调任务。

AI for OS

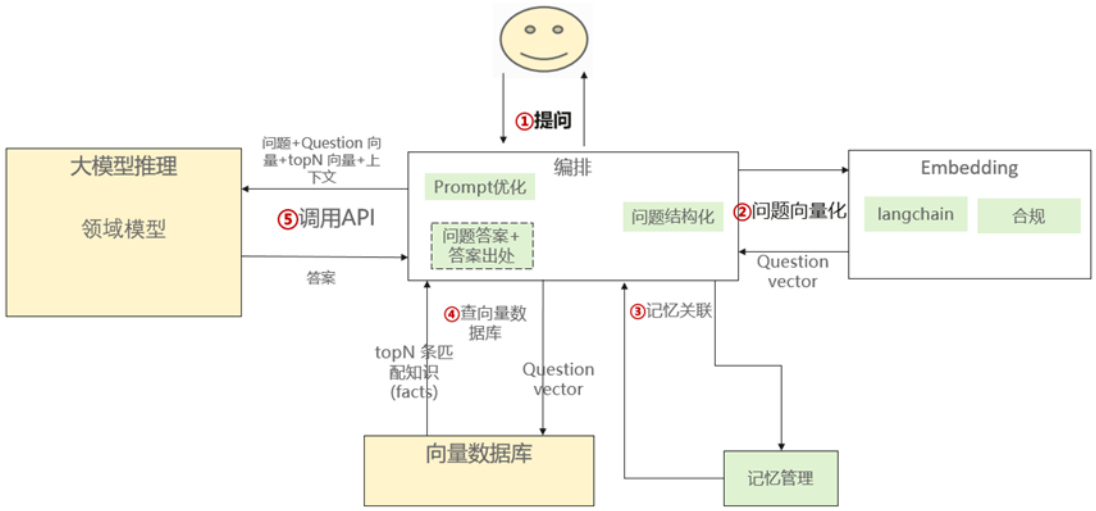
当前，openEuler 和 AI 深度结合，一方面使用基础大模型，基于大量 openEuler 操作系统的代码和数据，训练出 EulerCopilot，初步实现代码辅助生成、智能问题智能分析、系统辅助运维等功能，让 openEuler 更智能。

EulerCopilot 智能问答

功能描述

EulerCopilot 智能问答平台目前支持 web 和智能 shell 两个入口。

- Web 入口：操作简单，可咨询操作系统相关基础知识，openEuler 动态数据、openEuler 运维问题解决方案、openEuler 项目介绍与使用指导等等。
- 智能 Shell 入口：自然语言和 openEuler 交互，启发式的运维。



应用场景

- 面向 openEuler 普通用户：深入了解 openEuler 相关知识和动态数据，比如咨询如何迁移到 openEuler。
- 面向 openEuler 开发者：熟悉 openEuler 开发贡献流程、关键特性、相关项目的开发等知识。
- 面向 openEuler 运维人员：熟悉 openEuler 常见或疑难问题的解决思路和方案、openEuler 系统管理知识和相关命令。

相关使用方式请参考 EulerCopilot 智能问答用户指南。

openEuler Embedded

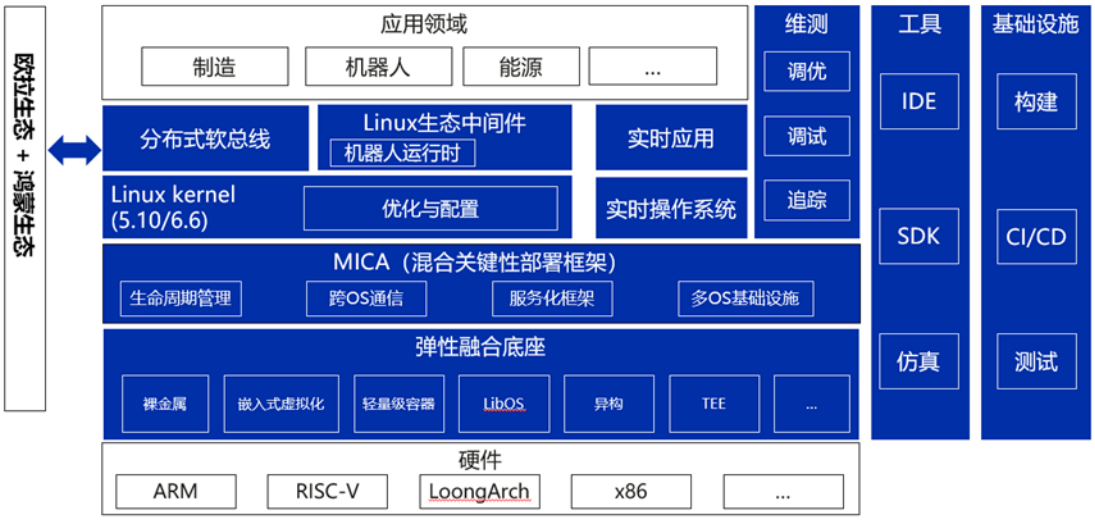
openEuler 发布面向嵌入式领域的版本 openEuler 24.03 LTS Embedded，构建了一个相对完整的综合嵌入系统软件平台，在南北向生态、关键技术特性、基础设施、落地场景等

方面都有显著的进步。

openEuler Embedded 围绕以制造、机器人为代表的 OT 领域持续深耕，通过行业项目垂直打通，不断完善和丰富嵌入式系统软件栈和生态。在内核方面，结合嵌入式场景的实际需求，支持 5.10 和 6.6 双内核，用户可以结合 BSP、应用场景等实际需求在两个版本的内核中选择其一，同时开发了自定义内核的能力。嵌入式弹性底座支持多种解决方案，包括 Jailhouse 分区虚拟化方案、openAMP 裸金属混合部署方案、基于 ZVM 和 Rust-Shyper 的实时虚拟化部署方案，用户可以根据自己的使用场景选择最优的部署方案。在嵌入式弹性底座之上打造了混合关键性部署框架 MICA，对下屏蔽不同底座的差异，对上为不同运行时提供统一的接口。在北向，目前已经支持 600+软件包，包括支持 ROS humble 版本，集成 ros-core、ros-base、SLAM 等核心软件包，满足 ROS2 运行时要求，针对嵌入式上层用户开发 SDK，加入了 ROS2 的嵌入式特色能力，SDK 支持 ROS2 colcon 交叉编译；支持 BMC 生态，已初步支持 openBMC。在南向，新增飞腾、海思、瑞萨、德州仪器、全志等硬件的支持，特别是提出了面向开发者的硬件开发板概念“欧拉派/Euler Pi”，并具体推出了第一款 openEuler Embedded 原生开发板“海鸥派/HiEuler Pi”。在基础设施，正式发布 openEuler Embedded 元工具 oebuild，构建系统升级到 Yocto 4.0 LTS，工具链新增支持 LLVM 工具链，可以采用 LLVM 工具链来构建镜像，并生成对应的 SDK，相对 GCC 工具链，可以获得在性能、体积、安全性等诸多方面的改进。在落地场景方面，openEuler Embedded 已经有了多个商业发行版/自用版，在 BMC，工业控制器，机器人控制器等领域开始应用。

未来 openEuler Embedded 将协同 openEuler 社区生态伙伴、用户、开发者，逐步扩展支持龙芯等新的芯片架构和更多的南向硬件，完善工业中间件、嵌入式 AI、嵌入式边缘、仿真系统等能力，打造综合嵌入式系统软件平台解决方案。

系统架构图



南向生态

openEuler Embedded Linux 当前主要支持 Arm64、x86-64、Arm32、RISC-V 等多种芯片架构，未来计划支持龙芯等架构，24.03 LTS 版本的南向支持大幅改善，已经支持树莓派、海思、瑞芯微、瑞萨、德州仪器、飞腾、赛昉、全志等厂商的芯片。

嵌入式弹性虚拟化底座

openEuler Embedded 的弹性虚拟化底座是为了在多核片上系统（SoC, System On Chip）上实现多个操作系统共同运行的一系列技术的集合，包含了裸金属、嵌入式虚拟化、轻量级容器、LibOS、可信执行环境（TEE）、异构部署等多种实现形态。不同的形态有各自的特点：

1. 裸金属：基于 openAMP 实现裸金属混合部署方案，支持外设分区管理，性能最好，但隔离性和灵活性较差。目前支持 UniProton/Zephyr/RT-Thread 和 openEuler Embedded Linux 混合部署。
2. 分区虚拟化：基于 Jailhouse 实现工业级硬件分区虚拟化方案，性能和隔离性较好，但灵活性较差。目前支持 UniProton/Zephyr/FreeRTOS 和 openEuler Embedded Linux 混

合部署，也支持 openHarmony 和 openEuler Embedded Linux 的混合部署。

3. 实时虚拟化：openEuler 社区孵化了嵌入实时虚拟机监控器 ZVM 和基于 rust 语言的 Type-I 型嵌入式虚拟机监控器 Rust-Shyper，可以满足不同场景的需求。

混合关键性部署框架

openEuler Embedded 打造了构建在融合弹性底座之上混合关键性部署框架，并命名为 MICA（MIxed CriticAlity），旨在通过一套统一的框架屏蔽下层弹性底座形态的不同，从而实现 Linux 和其他 OS 运行时便捷地混合部署。依托硬件上的多核能力使得通用的 Linux 和专用的实时操作系统有效互补，从而达到全系统兼具两者的特点，并能够灵活开发、灵活部署。

MICA 的组成主要有四大部分：生命周期管理、跨 OS 通信、服务化框架和多 OS 基础设施。生命周期管理主要负责从 OS（Client OS）的加载、启动、暂停、结束等工作；跨 OS 通信为不同 OS 之间提供一套基于共享内存的高效通信机制；服务化框架是在跨 OS 通信基础之上便于不同 OS 提供各自擅长服务的框架，例如 Linux 提供通用的文件系统、网络服务，实时操作系统提供实时控制、实时计算等服务；多 OS 基础设施是从工程角度为把不同 OS 从工程上有机融合在一起的一系列机制，包括资源表达与分配，统一构建等功能。

混合关键性部署框架当前能力：

- 支持裸金属模式下 openEuler Embedded Linux 和 RTOS（Zephyr/UniProton）的生命周期管理、跨 OS 通信。
- 支持分区虚拟化模式下 openEuler Embedded Linux 和 RTOS（FreeRTOS）的生命周期管理、跨 OS 通信。

北向生态

1. 北向软件包支持：600+嵌入式领域常用软件包的构建。
2. ROS 运行时：支持 ROS2 humble 版本，集成 ros-core、ros-base、SLAM 等核心包，并提供 ROS SDK，简化嵌入式 ROS 开发。
3. 软实时内核：提供软实时能力，软实时中断响应时延微秒级。

4. 分布式软总线基础能力：集成 OpenHarmony 的分布式软总线和 hichain 点对点认证模块，实现欧拉嵌入式设备之间互联互通、欧拉嵌入式设备和 OpenHarmony 设备之间互联互通。
5. 嵌入式容器与边缘：支持 iSula 容器，可以实现在嵌入式上部署 openEuler 或其他操作系统容器，简化应用移植和部署。支持生成嵌入式容器镜像，最小大小可到 5MB，可以部署在其他支持容器的操作系统之上。支持 Kubeedge，可以更好地实现“云-边-端”协同。

UniProton 硬实时系统

UniProton 是一款实时操作系统，具备极致的低时延和灵活的混合关键性部署特性，可以适用于工业控制场景，既支持微控制器 MCU，也支持算力强的多核 CPU。目前关键能力如下：

- 支持 Cortex-M、Arm64、x86_64、riscv64 架构，支持 M4、RK3568、RK3588、x86_64、Hi3093、树莓派 4B、鲲鹏 920、昇腾 310、全志 D1s。
- 支持树莓派 4B、Hi3093、RK3588、x86_64 设备上通过裸金属模式和 openEuler Embedded Linux 混合部署。
- 支持通过 gdb 在 openEuler Embedded Linux 侧远程调试。
- 支持 890+ POSIX 接口，支持文件系统、设备管理、shell 控制台、网络。

应用场景

openEuler Embedded 可广泛应用于工业控制、机器人控制、电力控制、航空航天、汽车及医疗等领域。

5. 内核创新

openEuler 内核中的新特性

openEuler 24.03 LTS 基于 Linux Kernel 6.6 内核构建，在此基础上，同时吸收了社区高

版本的有益特性及社区创新特性：

上游继承特性：

- 内存管理 folio 特性：Linux 内存管理基于 page（页）转换到由 folio（拉丁语 foliō，对开本）进行管理，相比 page，folio 可以由一个或多个 page 组成，采用 struct folio 参数的函数声明它将对整个（1 个或者多个）页面进行操作，而不仅仅是 PAGE_SIZE 字节，从而移除不必要复合页转换，降低误用 tail page 问题；从内存管理效率上采用 folio 减少 LRU 链表数量，提升内存回收效率，另一方，一次分配更多连续内存减少 page fault 次数，一定程度降低内存碎片化；而在 IO 方面，可以加速大 IO 的读写效率，提升吞吐。全量支持匿名页、文件页的 large folio，提供系统级别的开关控制，业务可以按需使用。对于 Arm64 架构，基于硬件 contiguous bit 技术（16 个连续 PTE 只占一个 TLB entry），可以进一步降低系统 TLB miss，从而提升整体系统性能。
- EEVDF 调度：EEVDF 全称“Earliest Eligible Virtual Deadline First“。EEVDF 调度器将调度时延作为任务调度的考虑因素之一，在保障任务运行时间分配公平的同时，优先在没有满足应得运行时间的任务中，选择任务 deadline 最近的任务，从而保障任务的调度时延， 解决了原有的 CFS 调度器只能公平分配任务运行时间，不能满足任务时延要求的问题。
- cgroup v2 特性：cgroup v2 相比 v1，具有统一的层级结构、更完善的线程模式管理、更安全的子树委派以及更丰富的特性支持。

  1）统一层级结构：

  简化 cgroup 的层级管理，用户不需要为不同的资源管理配置多个独立的 cgroup 树，降低多个控制器协同工作控制难度。提供了更一致和简化的接口，使得配置更简单易懂。更高的安全性，避免父子 cgroup 资源竞争：cgroup v2 新增只有父 cgroup 内部无进程时才能启用子 cgroup 控制器的限制。

  2）更完善的线程模式管理：

  cgroup-v2 引入线程模式（threaded），对可线程化管理的子系统进行限制。线程可以被独立于进程其他线程分配到不同的 cgroup 中，对单个线程的资源使用进行更精细的控制。

  3）更安全的子树委派：

  通过委派机制允许非特权用户创建和管理自己的 cgroup 层次结构。通过合理利用委

派，系统管理员可以提供给用户或应用程序必要的控制权限，提供更细粒度的资源管理，同时保持系统的稳定性和安全性。

4）更丰富的特性支持:

基于统一的文件树管理，支持 psi、页面缓存回写、跨多个资源的增强资源分配管理和隔离、统一核算不同类型的内存分配， MemoryQoS 等特性。

- Maple Tree 和 Per VMA Lock 特性：使用 Maple Tree 替代红黑树管理进程的地址空间，并使用 RCU 友好的设计以及 Per VMA Lock 等机制，可以减少锁争用问题，提升 page fault 的扩展性，可以加速应用启动等并发场景的性能。
- PCP high 自适应调节：对于不同工作负载，页面分配/释放的性能要求通常也是不同的。通过 PCP high 自适应的调节，可以自动调整每个 CPU 的页面集的大小以优化页面分配/释放性能，可以优化内核多并发构建等并发场景的性能。
- MGLRU 特性：多层级的 LRU 机制，更精确地识别页面的冷热信息，提升高内存压力场景下的系统性能，降低 OOM 的概率。
- DAMON 特性：轻量级内存访问监控框架，可在用户态实现虚拟地址或物理地址的访问监控，实现轻量精确的在线监控，助力性能提升。
- Memory-tiering 特性：内存分层特性，目标以最高效和成本效益的方式满足内存的使用需求，降低内存的使用成本。
- hugetlb vmemmap 特性：新增适配支持 Arm64 架构上的 hugetlb vmemmap 特性，节省内存管理结构的开销，降低内存底噪。
- huge vmalloc 特性：对于超过了 huge-page 的最小 size 的空间的 vmalloc/vmap 分配，将会尝试使用 huge page 而不是 base page，减少 tlb miss，显著提高了 vmalloc 使用场景的性能。
- 支持 memfd_secret 系统调用：Linux 上创建“秘密内存区域”的新接口，允许用户空间进程创建其他任何人（包括内核）都无法访问的内存范围，可以用于存储私钥等用途，减少在系统内存中暴露的可能性。
- BIG TCP 特性：允许协议栈发送更大的 TSO/GRO 数据包，实现更好的网络吞吐量性能和更低的延迟。
- XDP multi-buffer 支持：允许 Jumbo frame 场景下使用 XDP 提升性能。
- Thread-based NAPI polling 支持：允许网卡 NAPI polling 移到内核线程处理，使 CPU

调度器可以合理调度提升性能。

- bpf 新增 kfunc 特性：允许 bpf 通过符号方式直接调用内核和 ko 提供的函数，ko 可以通过注册 kfunc 的方式来动态提供 bpf 接口。
- bpf 新增 dynamic pointer 特性：在 bpf 中引用动态分配的内存，此前 bpf 使用的内存必须在 verifier 检查时已静态指定。
- perf 新增特性支持：新增支持 ARM SPE 事件的获取；支持用户态直接读取 pmu counter；显示锁的竞争情况；减少生成的 pmu 数据，提升平均处理时间。

openEuler 关键贡献：

- ext4 文件系统支持 Large folio：

  1）iomap 框架回写流程支持批量映射 block。

  2）支持 ext4 默认模式下批量申请 block，大幅优化各类 benchmark 下 ext4 性能表现。

  3）ext4 buffer io 读写流程以及 pagecache 回写流程弃用老旧的 buffer_head 框架，切换至 iomap 框架，并通过 iomap 框架实现 ext4 支持 large folio。
- 潮汐 affinity 调度特性：提供高效内存回收和加载机制，支持 per-memcg 的 swap 设备隔离，在业务负载低时回收冷数据，在业务负载升高时快速加载数据，实现内存可用空间和业务性能的提升，从而达到同等内存成本下更高的服务质量效果。

  感知业务负载动态调整业务 CPU 亲和性，当业务负载低时使用 prefered cpus 处理，增强资源的局部性；当业务负载高时，突破 preferred cpus 范围限制，通过增加 CPU 核的供给提高业务的 QoS。
- MPAM 特性：MPAM（内存系统组件隔离监控）是用于解决服务器系统中，混部不同类型业务时，由于共享资源的竞争而带来的关键应用性能下降或者系统整体性能下降的问题。另外，对于业务共享资源的使用，以 CPU 或者 PID 为粒度，进行实时跟踪监控。OLK 6.6 下 MPAM 重构的版本提供若干新特性：

  1）完整支持 L2 cache 分区隔离和监控功能，其中关于监控功能，支持统计缓存占用量，同时监控缓存带宽流量。

  2）支持任务之间根据优先级的差异，动态调整共享资源配置。

  3）支持任务共享资源的保底设置。

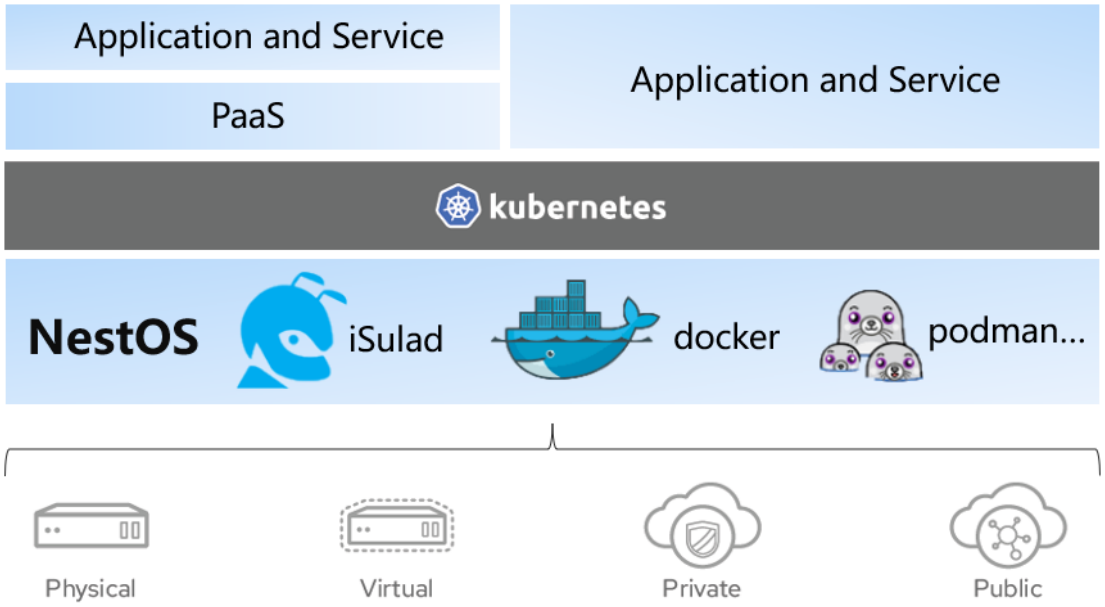
- CPU QoS 优先级负载均衡特性：在离线混部 CPU QoS 隔离增强，支持多核 CPU QoS 负载均衡，进一步降低离线业务 QoS 干扰。
- SMT 驱离优先级反转特性：解决混部 SMT 驱离特性的优先级反转问题，减少离线任务对在线任务 QoS 的影响。
- 混部多优先级：允许 cgroup 配置-2~2 的 cpu.qos_level，即多个优先级，使用 qos_level_weight 设置不同优先级权重，按照 CPU 的使用比例进行资源的划分，并提供唤醒抢占能力。
- 可编程调度：基于 eBPF 的可编程调度框架，支持内核调度器动态扩展调度策略，以满足不同负载的性能需求。
- 支持 arm64 vcpu hotplug：在线热插拔虚拟机 CPU 而不影响虚拟机正常运行，动态调整虚拟机的计算能力，提供安全可靠的虚拟机动态扩容。
- 负载算力协同：在多核服务器中运行用户体验敏感应用（如云桌面系统）时，通过负载算力协同技术能够保障算力供给的及时性和有效性。负载算力协同技术具有以下特性：高负载场景下，支持轻量级的任务搜索算法，提高空闲 CPU 拉取 runnable 任务的效率，实现多核间快速负载均衡，最大化 CPU 资源利用率；算力竞争场景下，支持按优先级对业务进行分级管控，有效避免优先级翻转的问题，实现高优先级的前台任务绝对压制低优级的后台任务，保障关键任务的算力供给。
- 已挂载文件系统设备写访问控制：支持块设备(分区设备)挂载文件系统后的写访问控制，为高危写访问操作提供告警信息，防止文件系统损坏。
- riscv bpf 新特性支持：支持 trampoline 特性、Zbb 扩展特性、kfunc 支持特性、cpu-v4 指令特性和原子操作指令特性等，扩展了 bpf 的能力。
- arm64 bpf 新特性支持：新增 bpf 栈变量 ldr/str 优化、pac 支持、trampoline 支持与 cpuv4 指令支持等特性，扩展了 bpf 的能力。
- 内存可靠性分级（继承特性）：支持使用者按照需求分配在对应可靠性的内存上，并对部分可能的 UCE 或 CE 故障影响进行一定程度的缓解，达到部分 MR 内存（address range mirror）的情况下，支撑业务整体可靠性不下降。

6. 云化基座

NestOS 容器操作系统

NestOS 是在 openEuler 社区孵化的云底座操作系统，集成了 rpm-ostree 支持、ignition 配置等技术。采用双根文件系统、原子化更新的设计思路，使用 nestos-assembler 快速集成构建，并针对 K8S、OpenStack 等平台进行适配，优化容器运行底噪，使系统具备十分便捷的集群组建能力，可以更安全的运行大规模的容器化工作负载。

功能描述



1. 开箱即用的容器平台：NestOS 集成适配了 iSulad、Docker、Podman 等主流容器引擎，为用户提供轻量级、定制化的云场景 OS。
2. 简单易用的配置过程：NestOS 通过 ignition 技术，可以以相同的配置方便地完成大批量集群节点的安装配置工作。
3. 安全可靠的包管理：NestOS 使用 rpm-ostree 进行软件包管理，搭配 openEuler 软件包源，确保原子化更新的安全稳定状态。
4. 友好可控的更新机制：NestOS 使用 zincati 提供自动更新服务，可实现节点自动更新与重新引导，实现集群节点有序升级而服务不中断。
5. 紧密配合的双根文件系统：NestOS 采用双根文件系统的设计实现主备切换，确保

NestOS 运行期间的完整性与安全性。

应用场景

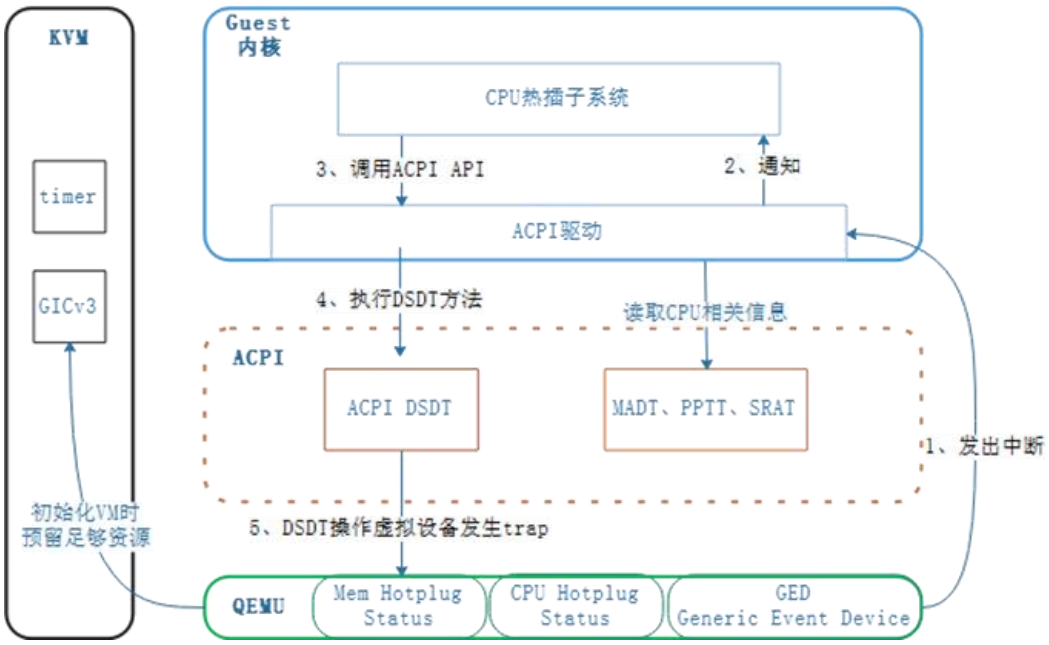
NestOS 适合作为以容器化应用为主的云场景基础运行环境，在本次 openEuler 24.03 LTS 版本中，引入社区孵化项目 NestOS-Kubernetes-Deployer，辅助 NestOS 解决在使用容器技术与容器编排技术实现业务发布、运维时与底层环境高度解耦而带来的运维技术栈不统一，运维平台重复建设等问题，保证了业务与底座操作系统运维的一致性。

7. 特性增强

AArch64 平台 vCPU 热插拔

vCPU 热插拔是指在虚拟机处于运行状态下，为虚拟机增加或减少 CPU 数量而不影响虚拟机业务正常运行的方案。当虚拟机内部业务压力不断增大，出现所有 CPU 均处于较高负载的情况下，可以使用 vCPU 热插功能提升虚拟机的计算能力，保障业务吞吐量稳定。当业务负载下降时，可以使用 vCPU 热拔功能去除多余的计算能力，从而降低计算成本。

功能描述



1. 特性增强：老版本已具备 vCPU 热插能力，该版本新增了对 vCPU 热拔的支持，适应更多使用场景。相对老版本，热插拔协议发生了一些变化，新老版本的热插拔不能兼容运行。即 Guest 内核版本和主机侧 QEMU 版本需配套才能实现热插拔功能。
2. 对外接口：保持和老版本一致，使用 libvirt setvcpus 接口完成热插拔操作。

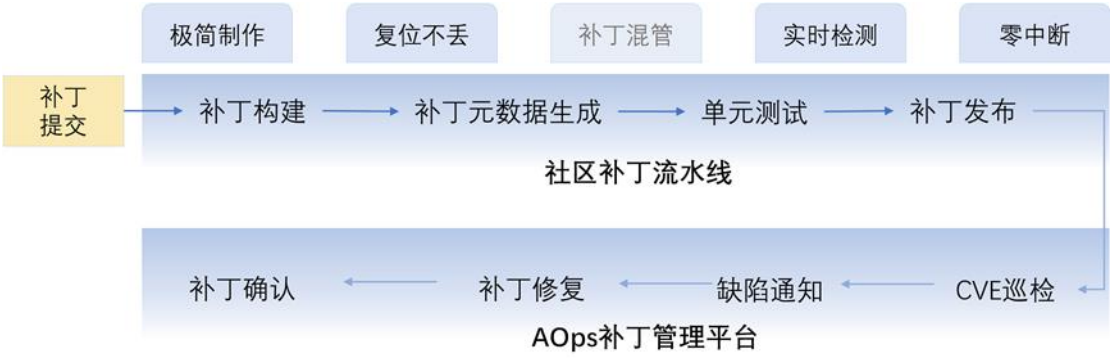
应用场景

1. 基于 vCPU 热插拔机制加快虚拟机启动速度。特别对于轻量安全容器场景收益较大。比如 Kata 安全容器初始只配置 1 个 vCPU，等启动完成后热插更多 vCPU。
2. 云厂商基于 vCPU 热插拔对用户按需扩展，用户根据业务负载需求，申请在线调整虚拟机 vCPU 数量，不影响业务运行。

A-Ops 智能运维

A-Ops 是一款基于操作系统维度的故障运维平台，提供从数据采集，健康巡检，故障诊断，故障修复到修复任务回退的智能运维解决方案。A-Ops 项目包括了诺干子项目：覆盖故障发现（aops-gala），故障定位支撑（aops-X-diagnosis），缺陷修复（aops-apollo）等。

本次发布的 aops-apollo 项目是智能补丁管理框架，集成了漏洞扫描、CVE 修复（冷补丁/热补丁）、热补丁回退等核心功能。系统可以对发布的安全公告实行定时下载同步，可设置定时任务执行漏洞扫描，保证系统平稳运行的同时，运维人员可通过 AOps 工具实现对漏洞的修复和回退。



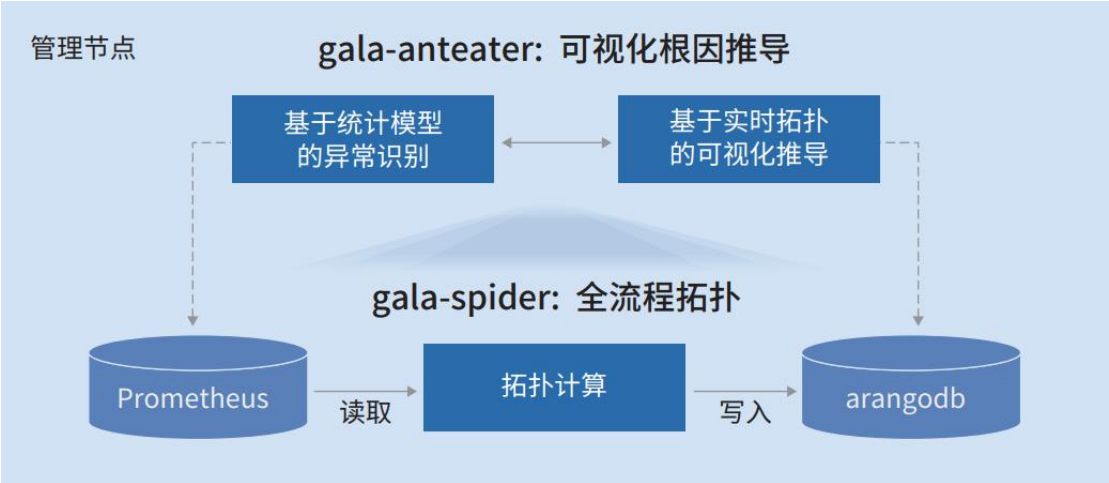
功能描述

aops-apollo 内核智能补丁管理

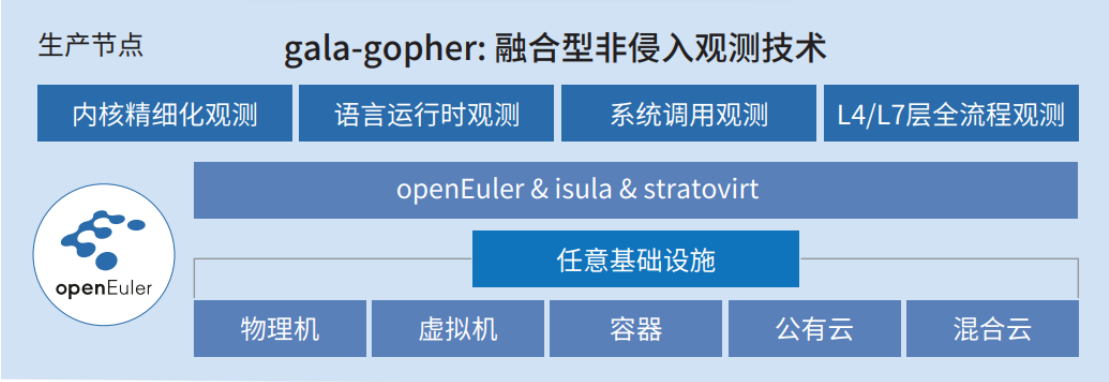
- 热补丁源管理：openEuler 的漏洞信息通过安全公告对外发布时，会同时在 update 源中发布修复所用的软件包。 默认 openEuler 系统安装后自带对应 OS 版本的冷补丁 update 源，用户也可以通过设置 repo 来自行配置冷/热补丁的 update 源。
- 缺陷扫描：通过对集群手动和定时扫描，检查集群是否受 CVE 影响，并提供冷/热补丁修复选择。
- 冷热补丁混合管理：支持冷补丁、热补丁独立修复，也支持冷热补丁混合修复，实现在网热补丁静默收编，减少热补丁维护成本。
- 热补丁生命周期管理：热补丁移除，回退，查询等生命周期管理。

aops-gala 项目

基于 eBPF + java agent 无侵入观测技术，并以智能化辅助，实现亚健康故障（比如性能抖动、错误率提升、系统卡顿等问题现象）诊断。其架构如图：



OpenTelemetry 生态接口



功能列表如下：

- 在线应用性能抖动诊断：提供数据库类应用性能在线诊断能力，包括网络类（丢包、重传、时延、TCP 零窗等）问题、 I/O 类（磁盘慢盘、I/O 性能下降等）问题，调度类（包括 sysCPU 冲高、死锁等）问题、内存类（OOM、泄漏等）问题等。
- 系统性能诊断：提供通用场景的 TCP、I/O 性能抖动问题诊断能力。
- 系统隐患巡检：提供内核协议栈丢包、虚拟化网络丢包、TCP 异常、I/O 时延异常、系统调用异常、资源泄漏、JVM 异常、应用 RPC 异常（包括 8 种常见协议的错误率、时延等）硬件故障（UCE、磁盘介质错误等）等秒级巡检能力。
- 系统全栈 I/O 观测：提供面向分布式存储场景的 I/O 全栈观测能力，包括 GuestOS 进程级、Block 层的 I/O 观测能力，以及虚拟化层存储前端 I/O 观测能力，分布式存储后端 I/O 观测能力。
- 精细化性能 Profiling：提供多维度（包括系统、进程、容器、Pod 等多个维度）、高精度（10ms 采样周期）的性能（包括 CPU 性能、内存占用、资源占用、系统调用等类型）火焰图、时间线图，可实时在线持续性采集。
- K8S Pod 全栈可观测及诊断：提供 K8S 视角的 Pod 集群业务流实时拓扑能力，Pod 性能观测能力、DNS 观测能力、SQL 观测能力等。

应用场景

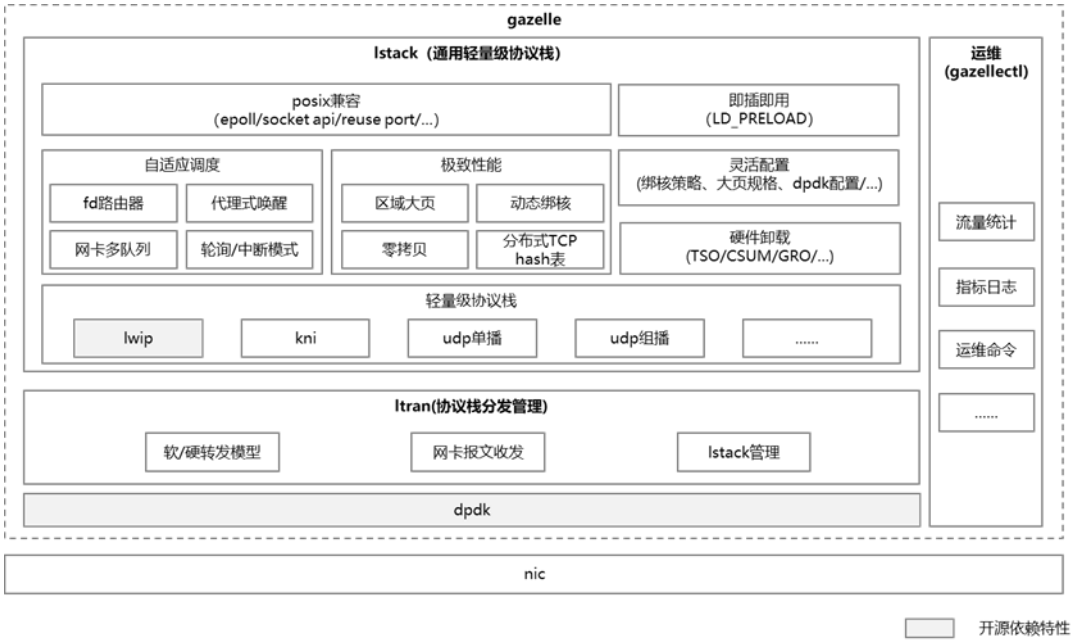
A-Ops 在 openEuler 等 Linux 环境主要面向场景包括数据库、分布式存储、虚拟化、云原生等场景。助力金融、电信、互联网等行业客户提供全栈可观测能力，能实现亚健康故障诊断；集群情况下人为导致的配置错误具备实时检查能力；冷热补丁混合管理能力，避免引入热补丁导致的补丁管理复杂。针对内核高分 CVE 直接提供热补丁，避免修复 Kernel 紧急

问题而需要重启系统。

Gazelle 特性增强

Gazelle 是一款高性能用户态协议栈。它基于 DPDK 在用户态直接读写网卡报文，共享大页内存传递报文，使用轻量级 LwIP 协议栈。能够大幅提高应用的网络 I/O 吞吐能力。专注于数据库网络性能加速，兼顾高性能与通用性。本次版本新增 UDP 协议及相关接口支持，丰富用户态协议栈。

功能描述



gazelle 功能架构图

- 高性能（超轻量）：基于 dpdk、lwip 实现高性能轻量协议栈能力。
- 极致性能：基于区域大页划分、动态绑核、全路径零拷贝等技术，实现高线性度并发协议栈。
- 硬件加速：支持 TSO/CSUM/GRO 等硬件卸载，打通软硬件垂直加速路径。
- 通用性（posix 兼容）：接口完全兼容 posix api，应用零修改，支持 udp 的 recvfrom 和 sendto 接口。
- 通用网络模型：基于 fd 路由器、代理式唤醒等机制实现自适应网络模型调度，udp 多

节点的组播模型，满足任意网络应用场景。

- 易用性（即插即用）：基于 LD_PRELOAD 实现业务免配套，真正实现零成本部署。
- 易运维（运维工具）：具备流量统计、指标日志、命令行等完整运维手段。

新增特性

- 新增支持单 vlan 模式、bond4 与 bond6 模式、网线插拔后网卡自愈功能。
- 全面支持鲲鹏 920 虚拟机单实例 redis 应用，最大支持链接数 5k+，性能提升约 30%+。
- 支持 netperf TCP_STREAM/TCP_RR（包长 1463 Byte 以下）参数测试。
- 对 gazelle 的 lstack、lwip、gazellectl 模块日志增强，便于定位。
- 支持 UDP 用户态协议栈，相比较内核协议栈性能提升约 50%。

应用场景

适用于网络 IO 是性能瓶颈的应用，对 Redis、MySQL 等数据库场景有较好的性能提升效果。

iSulad 特性增强

iSulad 是一个由 C/C++编写实现的轻量级容器引擎，具有轻、灵、易、快的特点，不受硬件规格和架构限制，底噪开销更小，可应用的领域更为广泛。 iSulad 采用统一的架构设计，支持云、边、端等多个场景，同时为不同场景提供不同的性能开销和内存底噪，以满足不同场景的容器化需求。本次版本新增 iSulad 对 CRI V1.29、cgroup v2 以及 CDI 的支持。

功能描述

- CRI V1.29：CRI（Container Runtime Interface，容器运行时接口）是 kublet 与容器引擎通信使用的主要协议。在本次版本中，iSulad 将支持 CRI V1.29，相比于 CRI v1.25，主要包含以下变更：

变更内容	变更描述
RuntimeConfig 接口	iSulad CRI V1 新增 RuntimeConfig 接口。iSulad 支持 systemd-cgroup 和 cgroupfs 两种驱动配置，用户可利用该

	接口获取 iSulad 的 cgroup 驱动配置类型。
GetContainerEvents 接口	iSulad CRI V1 新增 GetContainerEvents 接口。用户可以利用该接口通过流的方式获取 iSulad 中 pod 生命周期相关的事件。
ContainerStats 接口	ContainerStats 新增 SwapUsage 字段，用户可以通过 SwapUsage 字段获取虚拟内存使用信息。
ContainerStatus 接口	ContainerStatus 的 reason 字段新增 OOMKilled 值。用户可以通过 reason 字段判断容器是否在内存不足时发生了 OOM killed 事件。

- cgroup v2：cgroup 是 linux 中用于限制进程组资源的机制，目前包括两个版本，cgroup v1 和 cgroup v2。cgroup v2 相较于 cgroup v1 具有层次结构统一、资源控制精确、资源分配高效等特点。iSulad 在 cgroup v1 的基础之上，新增了对 cgroup v2 的支持，能够让 Kubernetes 委派更安全的 cgroup 子树给容器，支持跨多个资源的增强资源分配管理和隔离。
- CDI：CDI（Container Device Interface，容器设备接口）是容器运行时支持第三方设备的一种标准接口。设备供应商可以根据 CDI 规范，为设备编写设备描述文件，容器引擎可以根据描述文件对设备进行加载。iSulad 新增了对 CDI 的支持，支持用户加载符合 CDI 标准的设备。

约束限制

- CRI V1.29 的新增特性仅支持 runc 运行时
- 由于 cgroup oom 会同时触发容器 cgroup 路径删除，若 iSulad 对 oom 事件处理发生在 cgroup 路径删除之后，iSulad 则无法成功捕捉容器 oom 事件，可能导致 ContainerStatus 中 reason 字段设置不正确。
- iSulad 不支持交叉使用不同的 cgroup 驱动管理容器，启动容器后 iSulad 的 cgroup 驱动配置不应该发生变化。
- iSulad 只识别挂载在/sys/fs/cgroup 目录下的 cgroup。
- iSulad 不支持 cgroup v1 与 cgroup v2 混用场景，仅根据/sys/fs/cgroup 目录下的 cgroup 版本决定 iSulad 内部使用的 cgroup 版本。

- iSulad 目前仅支持 CRI 方式使用 CDI 特性。

应用场景

CRI V1.29 特性可用于 Kubernetes 1.29 版本对接 iSulad 的场景，用户可以通过 iSulad 的 enable-cri-v1 选项配置该特性。

iSulad 的 cgroup v2 特性可用于支持 cgroup v2 内核，只有内核打开了 cgroup v2 的特性，iSulad 对 cgroup v2 的支持才会生效。

iSulad 的 CDI 特性可用于支持兼容 CDI 规范的设备。用户在通过 CRI 创建容器时可以指定符合 CDI 规范的设备名称，iSulad 在创建容器时会将 CDI 描述文件中的设备正确挂载至容器内。

安全启动

安全启动（Secure Boot）是利用公私钥对启动部件进行签名和验证。在启动过程中，前一个部件验证后一个部件的数字签名，如果能验证通过，则运行后一个部件；如果验证不通过，则停止启动。通过安全启动可以保证系统启动过程中各个部件的完整性，防止没有经过认证的部件被加载运行，从而防止对系统及用户数据产生安全威胁。

openEuler 在支持安全启动的基础上，还通过支持内核模块签名、IMA 文件完整性保护等机制，将基于数字签名的保护链路进一步延伸至内核模块和应用程序文件（广义安全启动），整个验证过程可包含如下四个部分：

启动阶段：BIOS->shim->grub->内核（EFI 加载前进行签名校验）；

运行阶段（模块加载）：内核->内核模块（模块插入时进行签名校验）；

运行阶段（文件访问）：内核->文件（应用程序执行或普通文件访问时进行签名校验）；

运行阶段（软件包安装）：包管理组件->RPM 软件包（软件包安装时进行签名校验）。

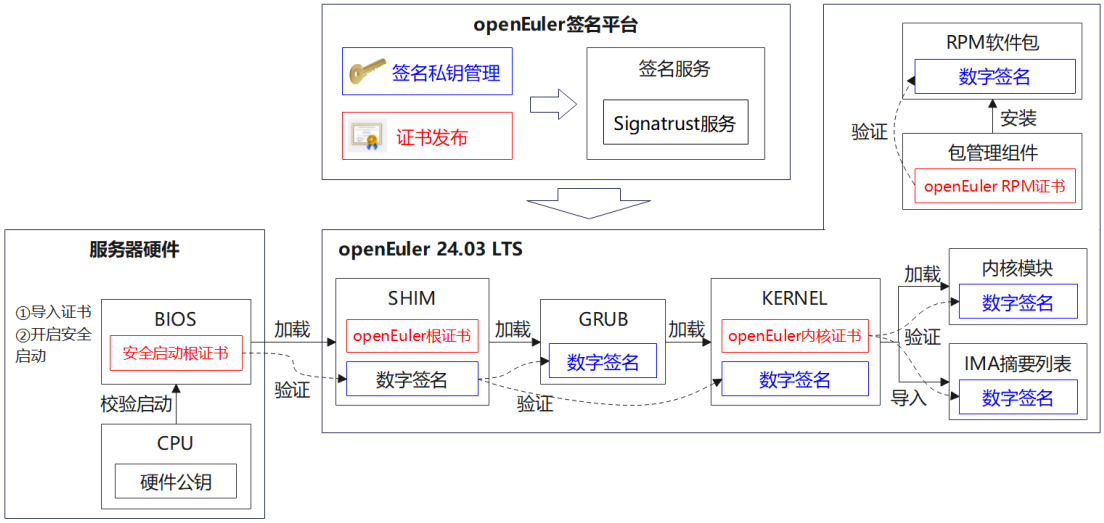
功能描述

为支持“开箱即用”的安全启动能力，核心是在 openEuler 社区建立以 PKI 为基础的软件构建签名体系，在软件构建阶段，自动为目标文件添加数字签名，并在关键组件中预置公钥

证书，从而在用户安装 openEuler 镜像后，可以直接开启相关的签名校验机制，提升系统安全性。

openEuler Signatrust 是社区基础设施 SIG 组针对操作系统常见的签名场景推出的高效、便捷、安全的签名服务，可支持 openPGP 及 X509 体系的密钥管理，同时对接了 EFI、RPM、KO、ISO 等多种目标文件格式，支持大批量的软件包签名，可极大提升社区密钥管理及软件包签名效率。

openEuler 24.03 LTS 版本的构建签名机制如下：



1. openEuler 签名平台生成并管理签名公私钥和证书，同时通过 Signatrust 提供签名服务；
2. EulerMaker 构建系统在执行软件包构建阶段，调用 Signatrust 签名接口为目标文件执行添加数字签名；
3. 具备签名验证功能的组件（如 shim、kernel 等），在构建阶段预置相应的验签证书；
4. 用户安装 openEuler 镜像后，开启安全启动、内核模块校验、IMA、RPM 校验等按机制，在系统启动和运行阶段使能相应的签名验证功能，保障系统组件的真实性和完整性。

openEuler 签名根证书可在证书中心获取：

https://www.openeuler.org/zh/security/certificate-center/

约束限制

- 当前社区签名平台只支持对 openEuler 社区内部构建的组件进行签名，暂不支持对外部工程构建的文件及客户文件进行签名。
- 当前签名平台提供的签名算法只支持国际 RSA 算法，密钥长度为 4096。

openEuler 24.03 LTS 版本已默认集成如下签名：

文件类型	文件格式	签名格式
EFI 文件	EFI Image	authenticode
内核模块文件	Kernel Module	CMS
IMA 摘要列表文件	Binary	CMS
RPM 软件包文件	RPM Package	openPGP

应用场景

用户可开启对应的安全机制，如安全启动、内核模块校验、IMA 等，开启后可使能签名校验功能。

GreatSQL 数据库

GreatSQL 数据库是一款开源免费数据库，可在普通硬件上满足金融级应用场景，具有高可用、高性能、高兼容、高安全等特性，可作为 MySQL 或 Percona Server for MySQL 的理想可选替换。

功能描述

- **高可用**

针对 MGR 进行了大量改进和提升工作，新增支持地理标签、仲裁节点、读写节点可绑定动态 IP、快速单主模式、智能选主，并针对流控算法、事务认证队列清理算法、节点加入&退出机制、recovery 机制等多项 MGR 底层工作机制算法进行深度优化，进一步提升优化了 MGR 的高可用保障及性能稳定性。

- 支持地理标签特性，提升多机房架构数据可靠性。
- 支持仲裁节点特性，用更低的服务器成本实现更高可用。
- 支持读写节点动态 VIP 特性，高可用切换更便捷。
- 支持快速单主模式，在单主模式下更快，性能更高。
- 支持智能选主特性，高可用切换选主机制更合理。

- 采用全新流控算法，使得事务更平稳，避免剧烈抖动。
- 优化了节点加入、退出时可能导致性能剧烈抖动的问题。
- 优化事务认证队列清理算法，高负载下不复存在每 60 秒性能抖动问题。
- 解决了个别节点上磁盘空间爆满时导致 MGR 集群整体被阻塞的问题。
- 解决了长事务造成无法选主的问题。
- 修复了 recovery 过程中长时间等待的问题。

更多信息详见文档：高可用。

- **高性能**

相对 MySQL 及 Percona Server For MySQL 的性能表现更稳定优异，支持高性能的内存查询加速 AP 引擎、InnoDB 并行查询、并行 LOAD DATA、事务无锁化、线程池等特性，在 TPC-C 测试中相对 MySQL 性能提升超过 30%，在 TPC-H 测试中的性能表现是 MySQL 的十几倍甚至上百倍。

- 支持类似 MySQL HeatWave 的大规模并行、高性能的内存查询加速 AP 引擎，可将 GreatSQL 的数据分析性能提升几个数量级。
- 支持 InnoDB 并行查询，适用于轻量级 OLAP 应用场景，在 TPC-H 测试中平均提升 15 倍，最高提升 40+倍。
- 优化 InnoDB 事务系统，实现了大锁拆分及无锁化等多种优化方案，OLTP 场景整体性能提升约 20%。
- 支持并行 LOAD DATA，适用于频繁导入大批量数据的应用场景，性能可提升约 20+倍。
- 支持线程池(Threadpool)，降低了线程创建和销毁的代价，保证高并发下，性能稳定不会明显衰退。

更多信息详见文档：高性能。

- **高兼容**

支持大多数常见 Oracle 用法，包括数据类型、函数、SQL 语法、存储程序等兼容性用法。

更多信息详见文档：高兼容。

- **高安全**

支持逻辑备份加密、CLONE 备份加密、审计日志入表、表空间国密加密等多个安全提升特性，进一步保障业务数据安全，更适用于金融级应用场景。

更多信息详见文档：高安全。

仓库地址

https://gitee.com/src-openeuler/greatsql

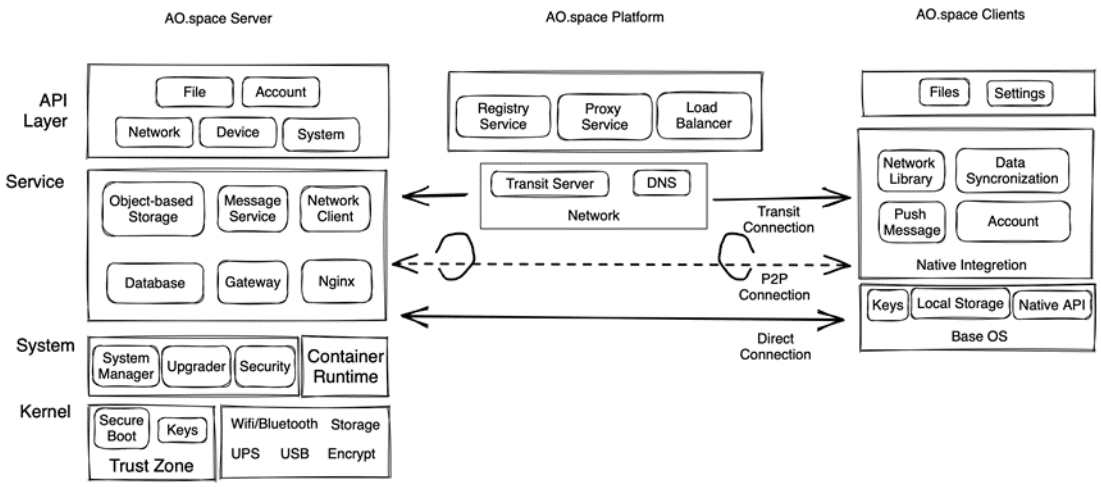
应用场景

GreatSQL 数据库常见的应用场景包括金融级高可用、高并发交易系统、Oracle 兼容迁移、高安全等级需求场景。

AO.space 项目发布

AO.space 是一个以保护个人数据安全和隐私为核心的解决方案。通过端对端加密、基于设备认证等技术，确保用户完全掌控个人账号和数据。同时，采用平台透明转发、点对点加速、局域网直连等技术，让用户随时随地的极速访问个人数据。利用 PWA 和云原生技术，设计并打造前后端一体的应用生态。

功能描述



AO.space 系统由三个主要部分构成：服务端、客户端和平台端。服务端为个人空间的核心，部署于个人长期运行的并且联网的设备中，如个人服务器、个人计算机等。客户

端为个人日常使用的电子设备，如手机、平板、个人电脑等，目前 AO.space 提供 Web、iOS 和 Android 等客户端。平台端在无法解析用户数据的前提下，为个人空间提供基础网络访问、安全防护等服务。

- **服务端：**服务端是 AO.space 的核心部分，一般部署在个人设备中，由空间软件、空间服务、容器运行时、基础操作系统（openEuler 等操作系统）和硬件组成。在基础操作系统之上，以容器方式部署空间的服务和应用，包括以下模块：
  - Web 服务（Nginx）：服务端的入口服务。
  - 代理（Agent）：既是空间基础服务的管理者，也是服务端、客户端与平台端之间沟通的桥梁，适应操作系统。
  - 网关（Gateway）：负责 API 的路由、转发、端到端加密和解密、认证以及整体空间应用层请求的授权。
  - AOFS：提供空间文件的存储和管理功能。它是一个虚拟文件系统，结合了对象存储和文件存储方法。
  - 预览（Preview）：负责为空间文件生成预览图。
  - 容器管理器（ContainerMgr）：用于与底层容器服务进行通信。
  - SQL 数据库实例（Postgresql）：为空间内的关系型数据库提供数据存储和管理。
  - NoSQL 数据库实例（Redis）：为空间内的非关系型数据库提供数据存储和管理，以及消息功能。
  - 网络客户端（Network client）：与平台端的网络转发服务建立安全通信通道，保证客户端与服务端在不同网络情况下的稳定通信。它还用于与客户端建立点对点（P2P）连接。
  - 空间应用：空间支持前端应用、后端应用和前后端混合应用三种类型，用于扩展空间功能。这些官方或第三方应用程序可以通过空间域名访问，例如 Card/CalDAV 服务。
- **客户端：**客户端是整个系统的前端，负责用户在不同的个人设备上与空间的交互，使用户能够随时随地访问空间的所有功能。目前提供 Web、iOS 和 Android 客户端，提供以下关键模块：
  - 端到端加密通道
  - 空间绑定

- 文件
- 设备
- 家庭
- 空间应用
- 安全

- **平台端：**平台提供基本网络资源和相关管理能力。它包括以下组件：
  - 入口网关（Endpoint）：负责处理和分配空间生态系统内的整体流量。
  - 基础服务（BaseService）：提供空间设备注册服务，同时协调和管理平台网络资源（域名、转发代理等）。
  - 网络转发服务（Transit server）：提供网络流量转发服务，使用户能够在大多数情况安全的通过互联网网络访问在办公室或家庭网络中的空间服务端。

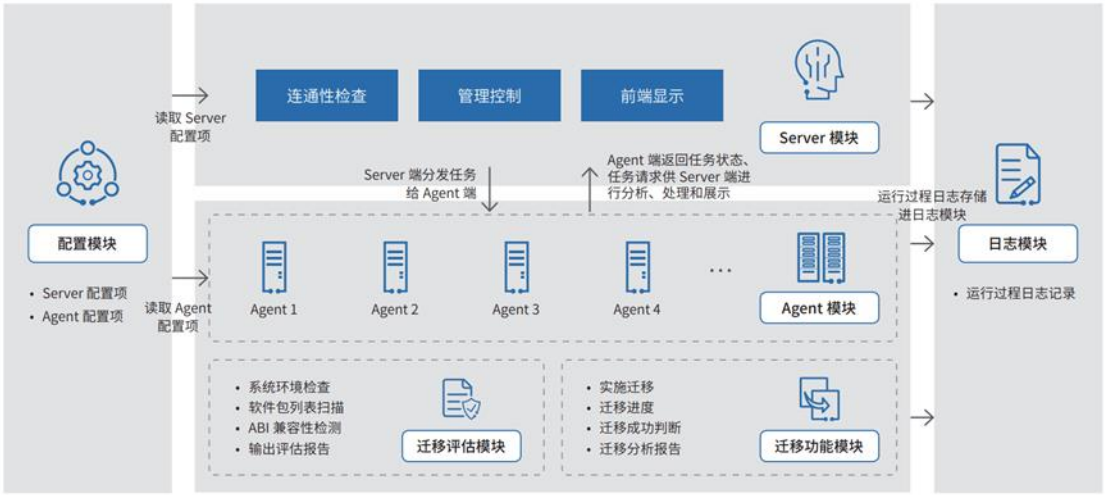
应用场景

在数字世界的快速发展中，AO.space 在个人自主数字身份、边缘计算、数据隐私保护、可信数据存储、个人 AI 助手等方面尝鲜设计和实践，促进个人数字生态的可持续发展。

Migration-tools 增强

Migration-tools 是由统信软件开发的一款操作系统迁移软件，面向已部署业务应用于其他操作系统且具有国产化替换需求的用户，帮助其快速、平滑、稳定且安全地迁移至 openEuler 系操作系统。现已支持在 web 端迁移 openEuler 操作系统。

功能描述



迁移软件的系统架构分为：Server 模块、Agent 模块、配置模块、日志模块、迁移评估模块、迁移功能模块。

- **Server 模块：**Server 模块为迁移的软件的核心，采用 pythonflaskweb 框架研发，负责接收任务请求，同时处理相关 执行指令并分发至各 Agent。
- **Agent 模块：**Agent 模块安装在待迁移的操作系统中，负责接收 Server 发出的任务请求，执行迁移等功能。
- **配置模块：**为 Server 模块和 Agent 模块提供配置文件的读取功能。
- **日志模块：**提供迁移的全部运行过程记录日志。
- **迁移评估模块：**提供迁移前的基础环境检测、软件包对比分析、ABI 兼容性检测等评估报告，为用户的迁移工作提供依据。
- **迁移功能模块：**提供一键迁移、迁移进度展示、迁移结果判断等功能。

应用场景

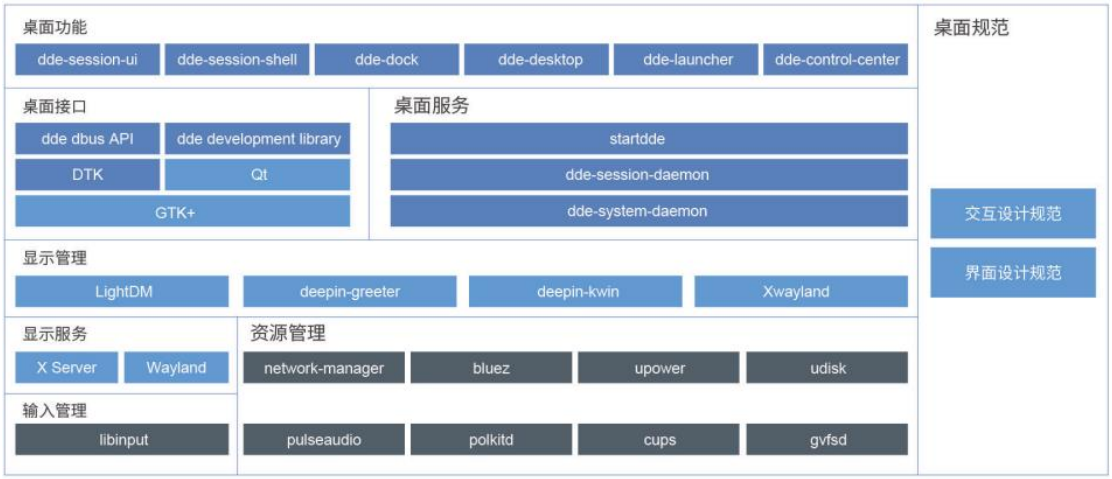
在金融、电信、能源等关键行业，涉及大量存量硬件设备（AMD64 等架构）中操作系统的国产化替代，需要将原存量 操作系统中的应用软件、系统组件迁移至 openEuler 操作系统中时，都可以使用 Migration-tools 进行迁移。

DDE 组件更新支持服务器场景

统信桌面环境（DDE）是统信软件为统信操作系统（UniontechOS）开发的一款桌面环境，统信桌面操作系统、统信操作系统服务器版和统信操作系统专用设备版均在使用统信桌面环境。

功能描述

统信桌面环境专注打磨产品交互、视觉设计，拥有桌面环境的核心技术，主要功能包含：登录锁屏、桌面及文件管理器、 启动器、任务栏（DOCK）、窗口管理器、控制中心等。由于界面美观、交互优雅、安全可靠、尊重隐私，一直是用户首选桌面环境之一， 用户可以使用它进行办公与娱乐，在工作中发挥创意和提高效率，和亲朋好友保持联系，轻松浏览网页、 享受影音播放。



统信桌面环境的核心技术是拥有统一界面元素设计、讲究细节交互设计的 DTK 框架及 Qt、GTK+ 等三方图形库。 显示服务、 输入管理、资源管理较为底层，一般是基于 golang 开发的后端服务，为上层 GUI 程序提供桌面环境中所需功能接口，如创建用户、设置屏幕亮度、设置设备音量、管理网络连接等功能。 显示管理、桌面接口、桌面服务属于 shell 层，一般是基于 DBus 接口协议 与后端服务进行通信，为定义用户界面、交 互操作提供支撑，如登录界面、窗口外观、GUI 应用程序控件等。

应用场景

桌面功能属于应用层，一般是面向用户可操作的功能界面，比如启动器、任务栏（DOCK）等。

kiran-desktop 2.6 版本

Kiran 是一款麒麟信安自研桌面环境产品，包含登录锁屏、开始菜单、控制中心等多种自研组件，采用模块化的设计风格，致力于提供界面友好、简单易用的人性化用户操作界面。该版本支持在 openEuler 中使用 Kiran2.6。

功能描述

Kiran 提供了操作系统的用户界面，包括了用户登录、桌面图标、控制面板、系统面板、文件管理器、桌面应用等功能。

- **用户登录：**Kiran 提供用户名和密码登录功能，在输入框输入用户名 -> 回车 -> 输入密码后回车进行登录。同时 Kiran 还具备多因子登录认证功能，用户可选择认证的类型有指纹、指静脉、Ukey、虹膜和人脸，在控制面板 -> 认证管理可录入信息，并配置是否启用。
- **桌面图标：**登录系统后，桌面图标说明如下。

  1) 计算机：双击可以显示从本计算机访问的所有本地和远程磁盘和文件夹。

  2) 主文件夹：双击可以显示用户家目录下的内容。

  3) 回收站：存放已删除的文件。
- **系统面板：**系统面板位于桌面下方区域，包括了任务栏、托盘区域和日期与时间。任务栏用于查看系统启动应用，默认放置开始菜单、文件管理器、Firefox 浏览器、工作区，也可通过鼠标右键点击应用，选择固定到任务栏添加其他应用到任务栏，可以进行应用程序打开、关闭、放大、最小化等操作。托盘区域可以设置输入法、调节音量、设置网络，托盘区域右边显示日期和时间。
- **控制面板：**控制面板是 Kiran 中提供的一个高度集成的图形化配置环境，几乎包含所有

的配置和管理工具，包括桌面定制、系统配置管理工具以及网络服务配置工具等。它主要包括以下功能：

1) 执行系统配置和管理任务。

2) 运行网络服务配置。

3) 定制具有个人特色的桌面环境。

- **文件管理器：**文件管理器是 Kiran 中提供的一款文件和目录管理工具，用户可通过文件管理器对系统中的文件与目录进行新增、编辑、复制、移动、删除、打开、剪切、重命名等操作。

应用场景

Kiran 提供了桌面常用的应用，如浏览器、终端、计算器、文本编辑器、磁盘和帮助手册等，点击开始菜单可看到操作系统中的所有应用。

- **Firefox 浏览器：**是一个自由及开放源代码网页浏览器，使用 Gecko 排版引擎，支持多种操作系统。它体积小速度快，还有其他一些高级特征，主要特性有：标签式浏览、使用网上冲浪更快、可以禁止弹出式窗口、自定制工具栏、扩展管理、更好的搜索特性、快速而方便的侧栏。
- **终端：**终端是操作系统使用系统命令操作的媒介，通过在终端窗口键入系统命令实现与系统交互。
- **计算器：**是一款快捷而简易的计算器，为用户提供加、减、乘、除等基本的数学计算。除了标准模式外，还提供了科学计算和程序员计算功能。
- **文本编辑器：**是一款快速记录文字的文档编辑工具，用于查看和修改纯文本文件。
- **磁盘：**是一款可查看、修改和配置磁盘与媒体的工具，可以通过该工具创建和恢复磁盘映像，也可对磁盘进行分区和格式化操作，
- **帮助手册：**Kiran 提供了帮助手册用于介绍系统中的桌面和常用应用功能，
- **截图工具：**是麒麟信安服务器操作系统自带的一款小巧灵活的屏幕捕捉软件，操作界面简洁、使用极为方便。

UKUI 支持

UKUI 是由麒麟团队开发的基于 Linux 发行版的轻量级桌面环境，将视觉和交互舒适自然的结合在一起，全面兼容 x86、Arm64 等多种主流架构，拥有更美观的 UI 界面、更友好的一致性交互体验，提供 4K 支持、夜间模式、任务栏预览等多种功能。

功能描述

- **控制面板：**进行系统的基本设置如日期与时间、个性化设置、设备管理等。
- **开始菜单：**管理系统中已安装的所有应用。可在默认和全屏尺寸两者之间切换，支持汉字、英文、拼音或首字母等多种检索方式。
- **任务栏：**支持深浅双色主题和毛玻璃效果，支持文件、文件夹、终端、网页、图片等窗口的实时预览。
- **侧边栏：**侧边栏一分为二，上半部分专管通知，下半部分提供了诸如截图、系统设置等快捷按钮。
- **文件管理器：**支持多标签页显示，将文件管理器的搜索栏和地址栏合二为一，重构的搜索功能，可以更加快速精准的搜索所需要的文件。

应用场景

UKUI 桌面为用户提供了简单高效的桌面图形环境。

OpenStack Wallaby、Antelope 多版本

OpenStack 是一个开源的云计算管理平台项目，旨在提供一个可扩展的、灵活的云计算服务，支持私有云和公有云环境。

功能描述

OpenStack 提供了一系列的服务和工具，用于构建和管理公共云、私有云和混合云。其主要功能包括:

- 计算服务：提供虚拟机的创建、管理和监控等功能。它允许用户快速创建、部署和销毁虚拟机和容器实例，从而实现对计算资源的灵活管理和高效利用。
- 存储服务：提供对象存储、块存储和文件存储等多种存储服务。块存储服务（如 Cinder）允许用户动态分配和管理持久性块存储设备，如虚拟机硬盘。对象存储服务（如 Swift）则提供了可扩展的、分布式的对象存储解决方案，用于存储大量非结构化数据。
- 网络服务：提供虚拟网络的创建、管理和监控等功能，包括网络拓扑规划、子网管理、安全组配置等，这使得用户能够轻松构建复杂的网络结构，并确保网络的安全性和可靠性。
- 身份认证服务：提供用户、角色和权限等身份管理功能，管理用户、角色和权限的访问控制。这使得用户能够安全地访问和管理云资源，并确保数据的机密性和完整性。
- 镜像服务：提供虚拟机镜像的创建、管理和共享等功能，包括创建、上传、下载和删除镜像。这使得用户能够轻松地创建和管理虚拟机镜像，并快速部署新的虚拟机实例。

编排服务：提供自动化部署和管理应用程序的功能，支持多个服务之间的协作和集成。通过编排服务（如 Heat），用户可以定义应用程序的部署模板，并自动执行相关的部署和管理任务。

应用场景

OpenStack 的应用场景主要包括以下几种情况：

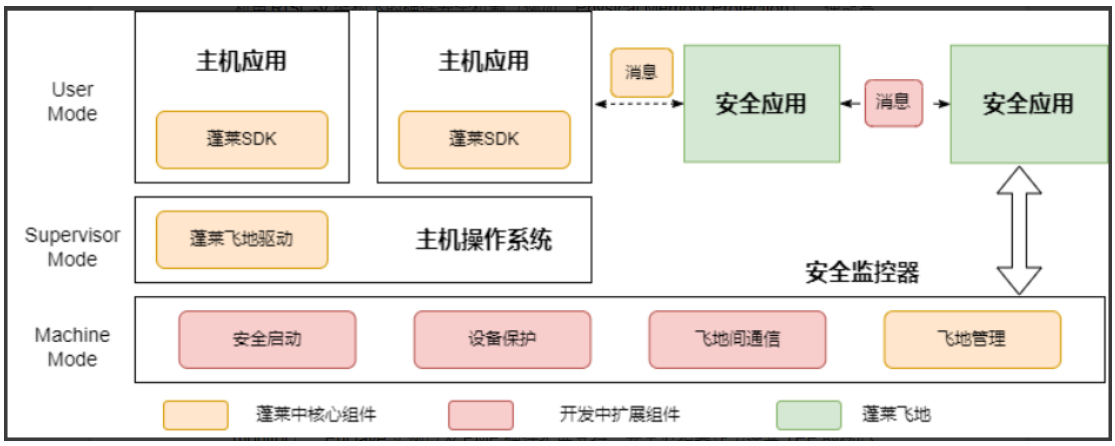
- 私有云：拥有私有云环境的企业可以根据自身的需求和 IT 资源的现状，选择 OpenStack 部署在自己的数据中心或云环境中。这可以实现资源集中管理、自动化部署和弹性扩展等功能，同时 OpenStack 还可以为企业提供强大的安全保障措施，如访问控制、数据加密和审计日志等。
- 公有云：OpenStack 也可以为公有云提供资源池，解决多租户云环境中不同用户之间的性能和隔离问题。在这种场景下，OpenStack 可以为公有云提供弹性计算、容器、网络和存储等基础设施服务，帮助公有云提供商实现资源集中管理和高可用性。
- 混合云：OpenStack 还可以为混合云提供一套完整的解决方案。混合云是指将私有云和公有云相互结合，以在云计算中获得更灵活、高效和安全的服务。混合云可以在私有云和公有云之间实现数据和应用的迁移、备份和恢复等操作。

- 大规模虚拟机管理：OpenStack 可以规划并管理大量虚拟机，从而允许企业或服务提供商按需提供计算资源。

RISC-V 架构 Penglai TEE 支持

蓬莱 TEE 补丁为 RISC-V OpenEuler 操作系统提供了可信执行环境（TEE）的支持，利用 RISC-V 架构下的硬件安全机制（例如：Physical Memory Protection），使能高安全性要求的应用场景：如安全通信、密钥保护、代码鉴权等。

功能描述



蓬莱 TEE 系统的主要组件包括开发工具 SDK、安全监控器（Secure monitor）、Enclave 实例以及 PMP 硬件扩展支持。安全监控器作为蓬莱 TEE 的核心，管理 Enclave 实例的生命周期和资源分配。Enclave 实例运行在用户态（U-mode），其内存受到 PMP 的保护。

约束限制：

- 当前蓬莱应用只支持 C 或者 C++代码。
- 对于 syscall 的支持依赖 secGear 中对 POSIX 接口的转发功能。
- 蓬莱系统实现需要 RISC-V 硬件支持 PMP 机制。
- 在资源受限的场景下，蓬莱 TEE 补丁当前能够同时支持 14 个 Enclave 实例，但具体数量可能受限于硬件资源，未来会扩展支持更多实例。
- 当前蓬莱 Enclave 默认使用 4MB 大小的安全内存，可以在蓬莱 Enclave Driver 中修改。
- 当前引入的蓬莱 TEE SDK 不包括 secGear 的代码，但用户可以参考 secGear 开发框架，自行下载与编译。

应用场景

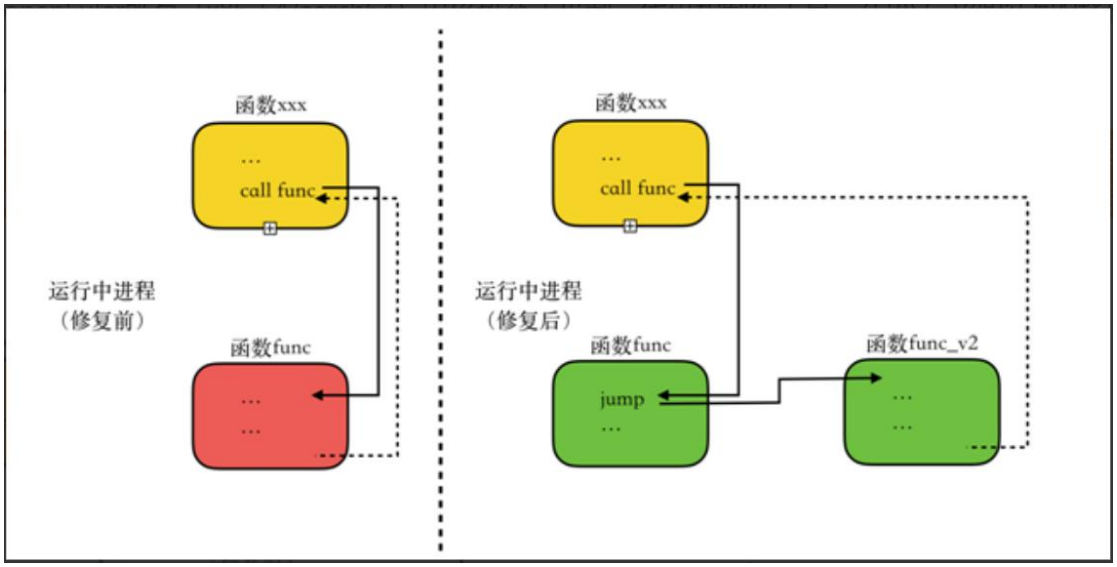
- 安全通信：在需要加密通信和数据保护的场合，如金融服务和军事通信，蓬莱 TEE 可以提供安全的加解密、签名验签以及哈希算法支持。
- 安全启动与鉴权：在设备启动和远程访问控制中，蓬莱补丁可以提供安全的启动流程和鉴权机制，防止未授权的访问和篡改。
- 数据加密处理：对于需要在加密环境下处理的数据，如个人隐私信息、企业敏感数据等，蓬莱 TEE 提供了一个安全的执行环境。
- 物联网设备：在物联网领域，蓬莱 TEE 的轻量级特性使其适合部署在资源受限的设备上，提供必要的安全保护。

RISC-V 架构内核热补丁能力

RISC-V 热补丁特性，是 openEuler 既有（x86-64/aarch64）内核热补丁机制、代码和制作工具，在 RISC-V 架构上的移植，功能、应用场景、约束等相同。 内核或应用程序不重启的情况下打补丁，是操作系统在服务器领域重要的性能和安全保障手段。

功能描述

以函数为基本修复单位，制作热补丁代码模块，在不重启机器/应用情况下，利用指令替换方式，将缺陷函数替换为修复函数，实现热补丁功能。主要包括内核热补丁机制（kernel）、内核态热补丁制作工具（kpatch）、用户态热补丁制作工具（libcareplus）、热补丁服务管理工具（syscare）。



约束限制：同 syscare、kpatch 等，主要包含以下几点。

- 仅支持 64 位系统。
- 仅支持 ELF 格式。
- 主要支持 C 语言。
- 有限支持修改全局/静态变量。
- 不支持多个文件、符号均同名。

应用场景

- 不可间断服务器内核缺陷热修复
- 不可间断应用服务缺陷热修复
- 操作系统发行厂商热补丁服务

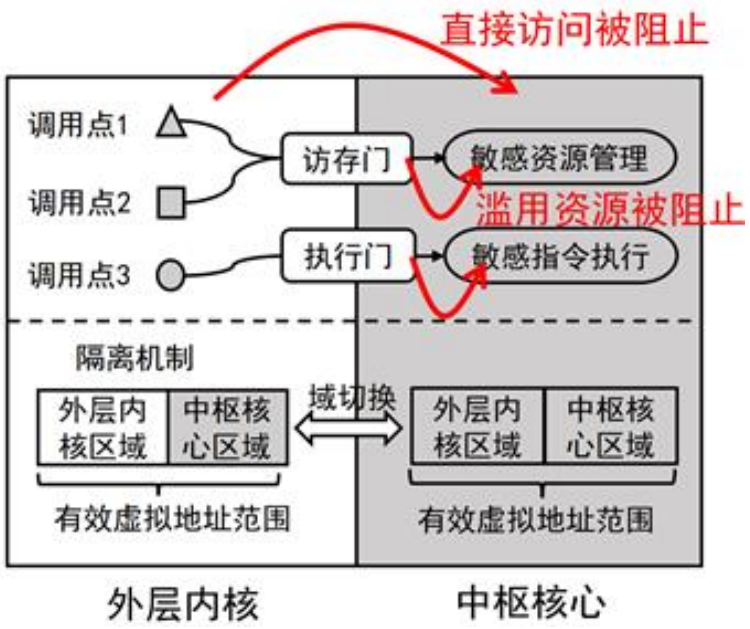
内核安全增强补丁 HAOC

HAOC (Hardware-Assisted OS Compartmentalization) 全称硬件辅助的操作系统分治技术，旨在通过对操作系统内部结构进行安全增强，以提升内核的抗攻击能力。HAOC 的核心技术是依据最小特权原则，对扁平化的 Linux 内核架构进行安全改造，形成核内再次分层的架构，即复式内核架构。复式内核架构基于现有处理器硬件机制在特权级内部实现多层次的系统隔离，能够有效阻止内核攻击的横向移动和权限提升能力。

复式内核架构的最内层，即中枢核心层，是复式内核架构的可信计算基（Trusted Computing Base，TCB），其可以在不信任 Linux 内核的前提下提供一个安全高效的隔离执行环境。中枢核心层不仅作为基础提供其他层次之间的隔离能力，而且还可以保护 Linux 核心内核中最安全敏感的资源，守住最后一道防线。

当前版本的 HAOC 技术，基于 Arm 处理器机制打造了复式内核架构的中枢核心层，并利用其保护了 Linux 内核的页表结构和进程凭证，可以有效缓解针对该类敏感资源的内核提权攻击。在未来 HAOC 将持续集成新功能，包括利用中枢核心层保护更多的敏感资源、构建复式内核高风险层实现设备驱动和内核扩展的安全管控、实现 x86 版本的复式内核架构等。

功能描述



HAOC 技术在 Linux 内核的地址空间内部构建了严格的隔离执行环境，即复式内核架构的中枢核心层。中枢核心基于 Arm 处理器 PAN 和 HPD 机制设计了访存门和执行门，分别实现了中枢核心的内存访问和代码执行隔离。访问中枢核心只允许通过安全门才可以：访存门用来阻止外层内核对中枢核心内存的直接读写访问，执行门用来阻止外层内核对中枢核心内敏感/特权指令的任意执行。安全门的设计不依赖外层内核的防护机制，其可以独立确保安全门执行的原子性和确定性。

中枢核心内部提供了不依赖 Linux 内核的完整运行环境，具备独立的堆栈和安全的中断

处理能力。当前中枢核心支持对内核页表和进程凭证进行安全保护，攻击者无法绕过安全门来修改这类敏感资源。为了进一步阻止潜在的混淆代理攻击，即攻击者滥用安全门接口非法使用敏感资源，中枢核心还对这类敏感资源的所有访问接口进行了安全检查。

应用场景

HAOC 特性可以从结构上提升 Linux 内核的抗攻击能力。即使内核存在未知漏洞，HAOC 依然可以阻止攻击者恶意篡改中枢核心内的敏感资源实现内核提权。

GCC for openEuler

GCC for openEuler 基线版本已经从 GCC 10.3 升级到 GCC 12.3 版本，支持自动反馈优化、软硬件协同、内存优化、SVE 向量化、矢量化数学库等特性。

1. GCC 版本升级到 12.3，默认语言标准从 C14/C++14 升级到 C17/C++17 标准，支持 Armv9-a 架构，x86 的 AVX512 FP16 等更多硬件架构特性。

	GCC 10.3.0	GCC 11.3.0	GCC 12.3.0
发布时间	2021/4/8	2022/4/21	2023/5/8
C标准	默认c17 支持c2x	默认c17 支持c2x	默认c17 支持c2x
C++标准	默认c++14 支持c++17 实验性C++2a改进 支持部分C++20	默认c++17 实验性C++2a改进 支持部分C++20	默认c++17 实验性C++2a改进 支持部分C++20
架构新特性	armv8.6-a (bfloat16 extension/Matrix Multiply extension) SVE2 Cortex-A77 Cortex-A76AE Cortex-A65 Cortex-A65AE Cortex-A34	armv8.6-a, +bf16, +i8mm armv8.6-r Cortex-A78 Cortex-A78AE Cortex-A78C Cortex-X1	armv8.7-a, +ls64 atomic load and store armv8.8-a, +mop, accelerate memory operations armv9-a Ampere-1 Cortex-A710 Cortex-X2 AVX512-FP16 SSE2-FP16

2. 支持结构体优化，指令选择优化等，充分使能 Arm 架构的硬件特性，运行效率高，在 SPEC CPU 2017 等基准测试中性能大幅优于上游社区的 GCC 10.3 版本。
3. 支持自动反馈优化特性，实现应用层 MySQL 数据库等场景性能大幅提升。

功能描述

- 支持 Arm 架构下 SVE 矢量化优化，在支持 SVE 指令的机器上启用此优化后能够提升程序运行的性能。
- 支持内存布局优化，通过重新排布结构体成员的位置，使得频繁访问的结构体成员放置于连续的内存空间上，提升 Cache 的命中率，提升程序运行的性能。
- 支持冗余成员消除优化，消除结构体中从不读取的结构体成员，同时删除冗余的写语句，缩小结构体占用内存大小，降低内存带宽压力，提升性能。
- 支持数组比较优化，实现数组元素并行比较，提高执行效率。
- 支持 Arm 架构下指令优化，增强 ccmp 指令适用场景，简化指令流水。
- 支持自动反馈优化，使用 perf 收集程序运行信息并解析，完成编译阶段和二进制阶段反馈优化，提升 MySQL 数据库等主流应用场景的性能。

应用场景

通用计算领域，运行 SPECCPU 2017 测试，相比于上游社区的 GCC 10.3 版本可获得 20%左右的性能收益。

其他场景领域，使能自动反馈优化后，MySQL 性能提升 15%以上；使能内核反馈优化后，实现 Unixbench 性能提升 3%以上。

7. 著作权说明

8. 商标

openEuler 白皮书上使用和显示的所有商标、标志皆属 openEuler 社区所有，但注明属于其他方拥有的商标、标志、商号除外。未经 openEuler 社区或其他方书面许可，openEuler 白皮书所载的任何内容不应被视作以暗示、不反对或其他形式授予使用前述任何商标、标志的许可或权利。未经事先书面许可，任何人不得以任何方式使用 openEuler 社区的名称及 openEuler 社区的商标、标记。

9. 附录

附录 1：搭建开发环境

环境准备	地址
下载安装 openEuler	https://openeuler.org/zh/download/
开发环境准备	https://gitee.com/openeuler/community/blob/master/zh/contributors/prepare-environment.md
构建软件包	https://gitee.com/openeuler/community/blob/master/zh/contributors/package-install.md

附录 2：安全处理流程和安全批露信息

社区安全问题披露	地址
安全处理流程	https://gitee.com/openeuler/security-committee/blob/master/security-process.md
安全披露信息	https://gitee.com/openeuler/security-committee/blob/master/security-disclosure.md