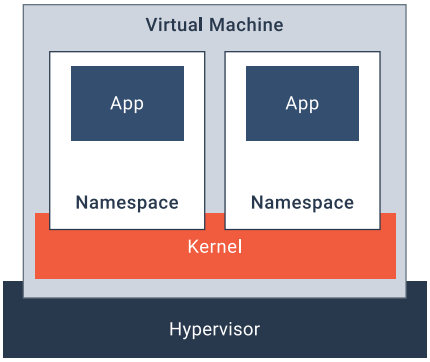
Kata Containers

Run AI and Sensitive Workloads with Hardware-Based Isolation

Kata Containers is a novel implementation of a lightweight virtual machine that seamlessly integrates within the container ecosystem. Lightweight & fast as regular containers, Kata Containers integrates with container management layers while also delivering the security advantages of virtual machines.

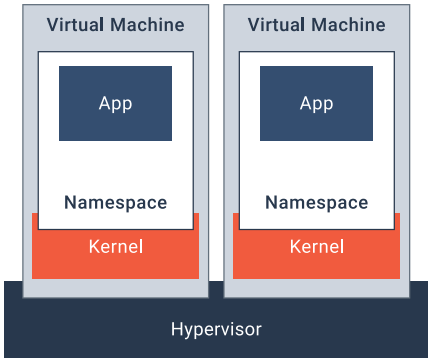
Born from the merger of two existing open source projects - Intel Clear Containers & Hyper runV - Kata Containers brings together the best of both technologies with a common vision of retooling virtualization to fit container-native applications, in order to deliver the speed of containers, and the security of VMs.

Kata Containers benefits from each project's strengths. Intel Clear Containers are focused on performance (<100ms boot time) and enhanced security, while Hyper runV prioritized being technology-agnostic by supporting many different CPU architectures and hypervisors. By merging the projects, Kata Containers delivers a superior end user experience in both performance and compatibility, unifies the developer communities, and accelerates feature development to tackle future use cases.



Containers in cloud today

(Shared kernel, isolation within namespace)



Kata Containers

(Each container/pod is hypervisor isolated, As secure as a VM, As fast as a container, Seamless integration with the container ecosystem and management layers)

With the increasing adoption of containers in enterprise environments, security and isolation are becoming more critical, and companies are looking for ways to ensure that their workloads are running in the most secure and isolated environment possible. Kata Containers utilizes open source hypervisors as an isolation boundary for each container (or collection of containers in a pod); this approach solves the shared kernel dilemma with existing bare metal container solutions.

Kata Containers is an excellent fit for both on-demand, event-based deployments such as serverless functions, continuous integration/continuous delivery, as well as longer running web server applications. The developer is no longer required to know anything about the underlying infra or to perform any kind of capacity planning before launching their container workloads. Kata Containers delivers enhanced security, scalability and higher resource utilization, while at the same time leading to an overall simplified stack.

Join The Community

Kata Containers is an independent open source community producing code under the Apache 2 license. Anyone is welcome to join and contribute code, documentation, and use cases.

katacontainers.io

Github
/kata-containers

OFTC IRC
#kata-dev

Slack
KataContainers

Website
katacontainers.io

Mastodon:
@KataContainers@openinfra.dev

Facebook
KataContainers

Mailing Lists
lists.katacontainers.io

Email
info@katacontainers.io

Blog:
https://katacontainers.io/blog

Check out the code

Supported by

OpenInfra FOUNDATION

Part of The Linux Foundation

Kata Containers

Run AI and Sensitive Workloads with Hardware-Based Isolation

katacontainers

Kata Containers Features:

Security	Runs in a dedicated kernel, providing isolation of network, I/O and memory and can utilize hardware-enforced isolation with virtualization VT extensions
Compatibility	Supports industry standards including OCI container format, Kubernetes CRI interface, as well as legacy virtualization technologies
Simplicity	Eliminates the requirement for nesting containers inside full blown VMs
Performance	Delivers consistent performance as standard Linux containers

Kata Containers Enables:

Multi-tenancy	Event-Driven Container-Native	Increased Resource efficiency	Bridge Ecosystems
Enables multiple tenants to share single container orchestration engine	Can be launched at anytime without any planning or pre-existing VM cluster requirement.	Small footprint allows for 10x increase in density compared to traditional VMs.	Utilizes both battle tested hypervisor and bleeding edge container technologies, providing an elegant and cohesive integration.

Kata Containers Use Cases:

"Kata Containers' features, such as network isolation with dedicated kernels and performance jitter, are critical for Ant Group's use case. Teams at Ant Group still share the same host kernel yet run different pods in different guest operating systems (OS), where the guest-OS-level isolation enabled by Kata Containers allows their service performance to be much more stable."

Ant Group

"Microsoft is leveraging Kata Containers along with sister project Confidential Containers to maintain zero-trust environments for their Azure Kubernetes Service offering. Zero-trust Cloud architecture is a security framework focused on maintaining security & protecting data by assuming that no device or user should be automatically trusted, even if they are within the network perimeter."

Microsoft

Securing AI Agents with Kata Containers

Agent Sandbox provides a flexible environment for running untrusted AI workloads across various runtimes, and leveraging it with **Kata Containers** introduces a vital layer of **hardware-based isolation**. Traditional containers share a host kernel, which means they remain susceptible to attacks and memory leakage through that shared resource; a compromised agent could potentially break out of its container and infect the host or neighboring tenants via that shared kernel. Kata Containers mitigates this by providing **zero host visibility** to the AI agent and strictly guarding host resources through hardware virtualization when using Kata Containers and Agent Sandbox together. This architecture ensures **memory isolation** and **zero persistence** between consecutive AI agent runs.