



Topics > Understanding Linux containers > What's a Linux container?

What's a Linux container?

Published May 11, 2022 • 4-minute read

[Copy URL](#)

Red Hat named a Leader in the 2023 Gartner® Magic Quadrant™

Red Hat was positioned highest for ability to execute and furthest for completeness of vision in the Gartner 2023 Magic Quadrant for Container Management.

[Read the report](#)



Overview

A Linux® container is a set of 1 or more processes that are isolated from the rest of the system. All the files necessary to run them are provided from a distinct image, meaning Linux containers are portable and consistent as they move from development, to testing, and finally to production.

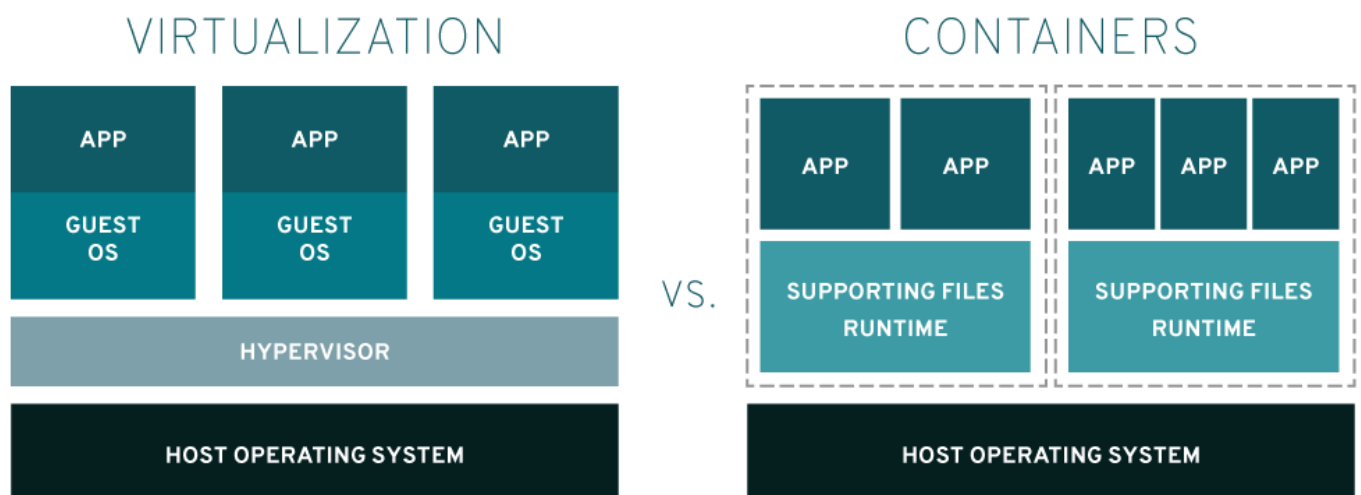
This makes them much quicker to use than development pipelines that rely on replicating traditional testing environments. Because of their popularity and ease of use containers are also an important part of IT security.

Start using Linux containers for free

Isn't this just virtualization?

Not exactly. Think of them more as complementary of one another. Here's an easy way to think about the 2:

- Virtualization lets your operating systems (Windows or Linux) run simultaneously on a single hardware system.
- Containers share the same operating system kernel and isolate the application processes from the rest of the system. For example: ARM Linux systems run ARM Linux containers, x86 Linux systems run x86 Linux containers, x86 Windows systems run x86 Windows containers. Linux containers are extremely portable, but they must be compatible with the underlying system.



What does this mean? For starters, virtualization uses a hypervisor to emulate hardware, which allows multiple operating systems to run side by side. This isn't as lightweight as using containers. When you have finite resources with finite capabilities, you need lightweight apps that can be densely deployed. Linux containers run natively on the operating system, sharing it across all of your containers, so your apps and services stay lightweight and run swiftly in parallel.

Linux containers are another evolutionary leap in how we develop, deploy, and manage applications. Linux container images provide portability and version control, helping ensure that what works on a developer's laptop also works in production. A special type of container image called a golden image creates a consistent, reliable baseline for system configuration. Compared to virtual machines, a running Linux container is less resource-intensive, has a standard interface

(start, stop, environment variables, etc.), retains application isolation, and is more easily managed as part of a larger application (multiple containers). Plus, those multi-container applications can be orchestrated across multiple clouds.

There are even tools that combine container orchestration and virtual machine management. Learn more about that in this Red Hat Summit 2020 track, which includes a breakout session on just such a tool.

Learn more about virtualization →

What is LXC?

The Linux Containers project (LXC) is an open source container platform that provides a set of tools, templates, libraries, and language bindings. LXC has a simple command line interface that improves the user experience when starting containers.

LXC offers an operating-system level virtualization environment that is available to be installed on many Linux-based systems. Your Linux distribution may have it available through its package repository.

A brief history of containers

The idea of what we now call container technology first appeared in 2000 as FreeBSD jails, a technology that allows the partitioning of a FreeBSD system into multiple subsystems, or jails. Jails were developed as safe environments that a system administrator could share with multiple users inside or outside of an organization.

In 2001, an implementation of an isolated environment made its way into Linux, by way of Jacques Gélinas' VServer project. Once this foundation was set for multiple controlled userspaces in Linux, pieces began to fall into place to form what is today's Linux container.

Very quickly, more technologies combined to make this isolated approach a reality. Control groups (cgroups) is a kernel feature that controls and limits resource usage for a process or groups of processes. And systemd, an initialization system that sets up the userspace and manages their processes, is used by cgroups to provide greater control over these isolated processes. Both of these technologies, while adding overall control for Linux, were the framework for how environments could be successful in staying separated.

Enter Docker

In 2008, Docker came onto the scene (by way of dotCloud) with their eponymous container technology. The docker technology added a lot of new concepts and tools—a simple command

line interface for running and building new layered images, a server daemon, a library of pre-built container images, and the concept of a registry server. Combined, these technologies allowed users to quickly build new layered containers and easily share them with others.

There are 3 major standards to ensure interoperability of container technologies—the OCI Image, Distribution, and Runtime specifications. Combined these specifications allow community projects, commercial products, and cloud providers to build interoperable container technologies (think pushing your custom built images into a cloud provider's registry server - you need that to work). Today Red Hat and Docker, among many others, are members of the Open Container Initiative (OCI)—are enabling an open, industry standardization of container technologies.

What's Docker →

What about container security?

Containers are popular, but how safe are they? There are a lot of moving parts to container security—you need to protect the container pipeline and application; the deployment environment(s) and infrastructure, and you need a plan for integrating with enterprise security tools and policies. You need a plan. We can help.

Read more on container security →

Why choose Red Hat for Linux containers?

Red Hat has a long history of working in the open source community to make technologies—like containers—secure, stable, and reliable. It's what we do. Then we support those technologies. So if you need help, we're there.

Red Hat's technologies take all of the guesswork out of doing containers the right way. Whether it's getting your development teams on a platform built with containers in mind, running your container infrastructure on a best-in-class operating system, or providing storage solutions for the massive data generated by containers, Red Hat's solutions have you covered.

See why you should trust us with your containers →

Keep reading

ARTICLE

What's a Linux container?

A Linux container is a set of processes isolated from the system, running from a distinct image that provides all the files necessary to support the processes.

Read more [→](#)

ARTICLE

Containers vs VMs

Linux containers and virtual machines (VMs) are packaged computing environments that combine various IT components and isolate them from the rest of the system.

Read more [→](#)

ARTICLE

What is container orchestration?

Container orchestration automates the deployment, management, scaling, and networking of containers.

Read more [→](#)

More about containers

Products	>
Related articles	>
Resources	>
Training	>




Products
Tools
Try, buy, & sell
Communicate

About Red Hat

We’re the world’s leading provider of enterprise open source solutions—including Linux, cloud, container, and Kubernetes. We deliver hardened solutions that make it easier for enterprises to work across platforms and environments, from the core datacenter to the network edge.

Select a language

 English ▼



- About Red Hat
- Jobs
- Events
- Locations
- Contact Red Hat
- Red Hat Blog

[Diversity, equity, and inclusion](#)

[Cool Stuff Store](#)

[Red Hat Summit](#)

© 2024 Red Hat, Inc.

[Privacy statement](#)

[Terms of use](#)

[All policies and guidelines](#)

[Digital accessibility](#)

[Cookie preferences](#)