Linux containers:

A container is a lightweight application isolation mechanism that allows the kernel to run groups of processes in their own isolated user spaces, separate from the host system.

The container has its own process list, network stack, file systems, and other resources, but shares the kernel with the host and the other containers running on the system.

Linux containers are implemented through a combination of three kernel features: namespaces for isolation, control groups for resource control, and SELinux for security.

A tool called Docker is used to create, control, and manage containers.

Docker adds an API, an image format, and a delivery and sharing model to Linux containers.

A Docker image contains an application and all its dependencies.

When a container is started, a read-write layer for that container is combined with the read-only image using LVM thin provisioning.

Docker images are portable and can be saved and exported to other systems and users.

Docker

Docker provides the user interface, API, image format, and other tools used to manage Linux containers in RHEL 7.

The docker package is provided by a special repository, RHEL 7 Extras (rhel-7-server-extras-rpms), which is updated under more aggressive engineering criteria than the core distribution.

Docker runs a daemon started by the systemd unit docker. service to manage containers.

The user interacts with this daemon through the docker command.

Images are stored in a local index kept in the /var /lib/docker directory, but are loaded and exported with the docker command.

Docker images

A Docker image is a static snapshot of a container's configuration, which is used to launch a container. The image is a read-only layer that is never modified. Instead, Docker adds a readwrite overlay to which all changes are made.

Changes are saved by creating a new image.

A single image can be used to generate many containers that are very slightly different, but only need enough disk space to store a very small amount of differences.

A platform image is an image with no parent. This is a baseline image that defines the core runtime environment, packages, and utilities needed to run an application.

Red Hat provides an extremely minimal platform image for RHEL 7 that has the minimum required packages to support a container, bash, and to use yum to download and install additional packages into a container.

The Docker registry

Platform images can be provided as a tar archive and loaded manually. More frequently, they are pulled from a Docker registry. This repository may be available to the public. Red Hat provides

Containers and virtualization

a read-only registry at registry. access. redhat. com, from which the RHEL 7 base platform image can be pulled. Docker maintains a public registry from which open source communityprovided images can be pulled.

It is also possible to use the docker-registry package to set up a simple private repository. Images can be pushed to this private repository and pulled by other machines in the network.

Containers and virtualization

Containers and virtualization are two technologies that provide complementary ways of dividing up system resources.

Containers can be run on virtual machines and in cloud computing environments, combining both technologies.

If virtualization "vertically" abstracts hardware, containers can be said to "horizontally" segment operating systems