Red Hat Summit

Connect

# Red Hat Enterprise Linux 10

Reimagine Linux


Red Hat
Enterprise Linux 10


# Alessandro Rossi

RHEL Advocate
alerossi@redhat.com



# Juan van der Breggen

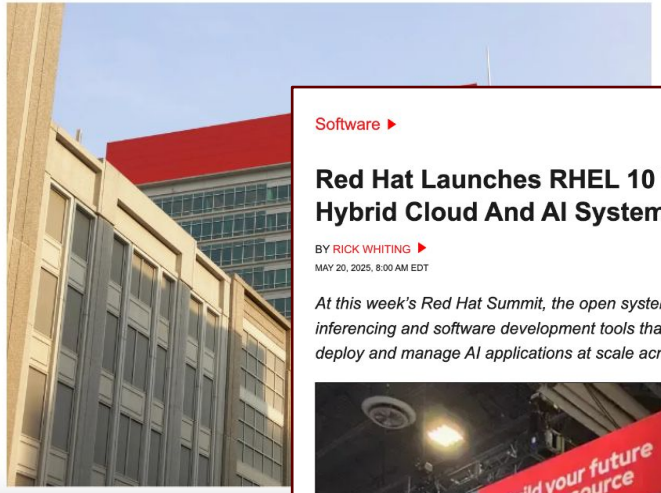
RHEL Advocate
juanb@redhat.com

Home / Business / Enterprise Software

# How Red Hat just quietly, radically transformed enterprise server Linux

**RHEL 10 becomes the first major enterprise Linux distro to discard traditional packaging and embrace immutable. See how we got here.**

Written by **Steven Vaughan-Nichols,** Senior Contributing Editor
June 2, 2025 at 11:02 a.m. PT



/ related

Software ▸

## Red Hat Launches RHEL 10 With New Capabilities For Hybrid Cloud And AI Systems

BY RICK WHITING ▸
MAY 20, 2025, 8:00 AM EDT

*At this week's Red Hat Summit, the open systems software giant also debuted new AI inferencing and software development tools that the company said will make it easier to bu deploy and manage AI applications at scale across hybrid ecosystems.*


theCUBE Coverage from SiliconANGLE's livestreaming video studio

UPDATED 11:05 EDT / MAY 23 2025


INFRA

## From AI to quantum-ready security, RHEL 10 brings the future to enterprise IT

BY BRIAN NJUGUNA

Red Hat Enterprise Linux continues to distinguish itself as a trusted business operating system, delivering enterprise-ready infrastructure tailored for mission-critical workloads. Now, RHEL 10 is emerging as the next-generation foundation for innovation and reliability in modern IT environments.

RHEL 10 introduces comprehensive enhancements in security, artificial intelligence integration,

InfoWorld

# Red Hat Enterprise Linux 10 adds AI-powered management

News
May 21, 2025 • 2 mins

Cloud Management | Configuration Management | Generative AI

Lightspeed is an AI-powered service that allows users to build, deploy, and manage Red Hat's Linux using simplified commands.


Credit: Michael Vi/Shutterstock

TECHZINE EU

3 MIN • INFRASTRUCTURE

## Red Hat launches RHEL 10 with intelligence and security for hybrid cloud

BERRY ZWETS
20 May 2025, 14:45 CEST

**Red Hat has released version 10 of Red Hat Enterprise Linux (RHEL). This new platform offers companies an intelligent and secure foundation for hybrid cloud environments and AI workloads. RHEL 10 introduces AI-driven management technology, post-quantum cryptography, and a container-native approach to operating systems and applications.**

One of the most notable innovations in RHEL 10 is the integration of AI into the management of Linux environments. With the new Red Hat Enterprise Linux Lightspeed functionality, administrators can access generative AI built directly into the operating system. This system provides context-aware guidance and recommendations through a

# Red Hat Enterprise Linux 10 will help you...


**Address the Linux skills gap**
with decades of Red Hat's Linux knowledge and expertise

**Contain drift and accelerate delivery**
with container tools and technologies

**Make better decisions at build time**
when it's typically easier and cheaper to make changes

**Resist security attacks from hackers**
when quantum computers become prevalent

**Leverage Red Hat Enterprise Linux as a trusted AI foundation**
with an extensive ecosystem of trusted partners and tools

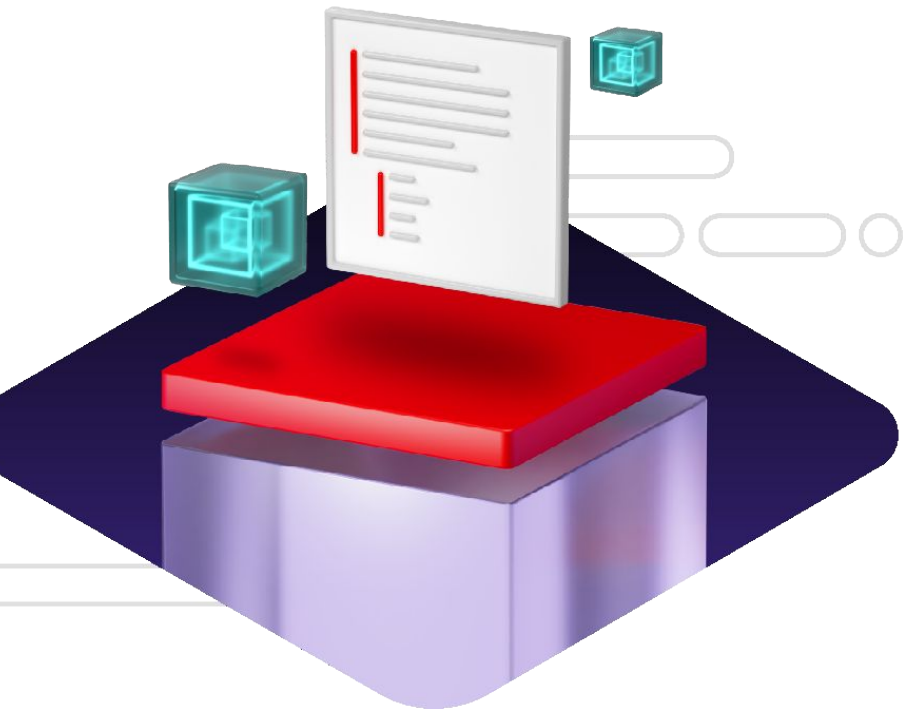
**Deploy workloads in the cloud faster**
with images pre-configured for performance on your cloud of choice



Red Hat

# Contain drift and accelerate delivery

## using container tools and technologies



## With image mode for Red Hat Enterprise Linux, you can:

- **Speed time to market**
  using DevOps and CI/CD practices, which now include the OS
- **Streamline operations**
  by automating updates and rollbacks—just like your smartphone
- **Enhance security**
  by reducing your attack surface with immutable system images
- **Simplify appliance creation**
  by combining the OS with apps and drivers for faster development and delivery

Because systems should be as easy to update as smartphones

Red Hat

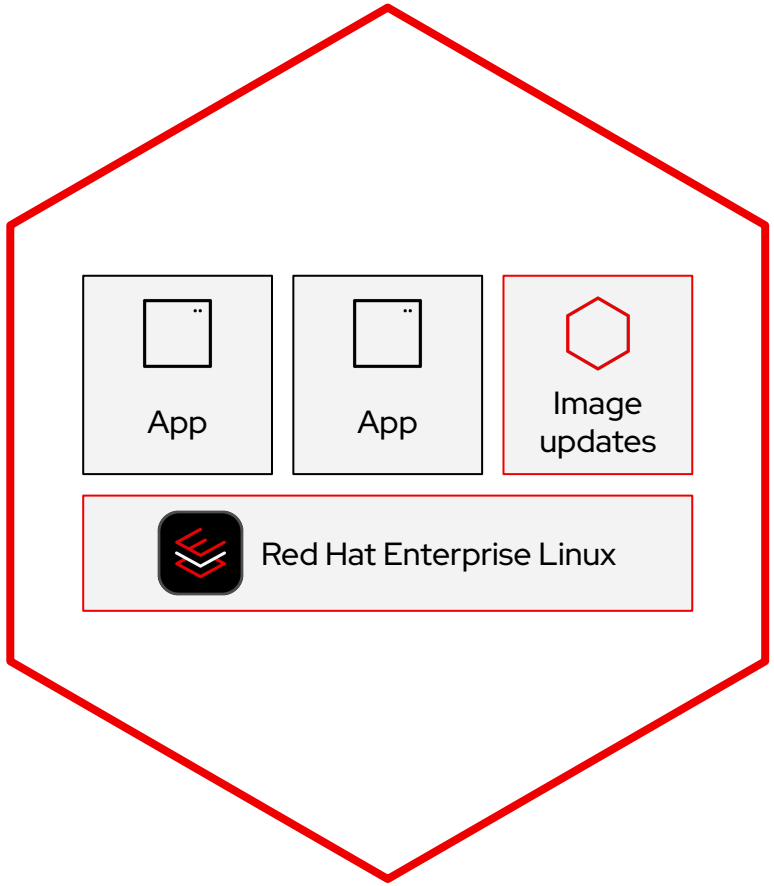
# “I want it to work like my smartphone”

## Factors that drove the search for a new solution

- Pain over regulatory process of CVEs
- Smaller footprint | decreased surface area of attack
- Enhanced security | hardened platform
- Quick turn around | lower downtime
- Easy rollbacks

FSI early adopter
Image mode for Red Hat Enterprise Linux

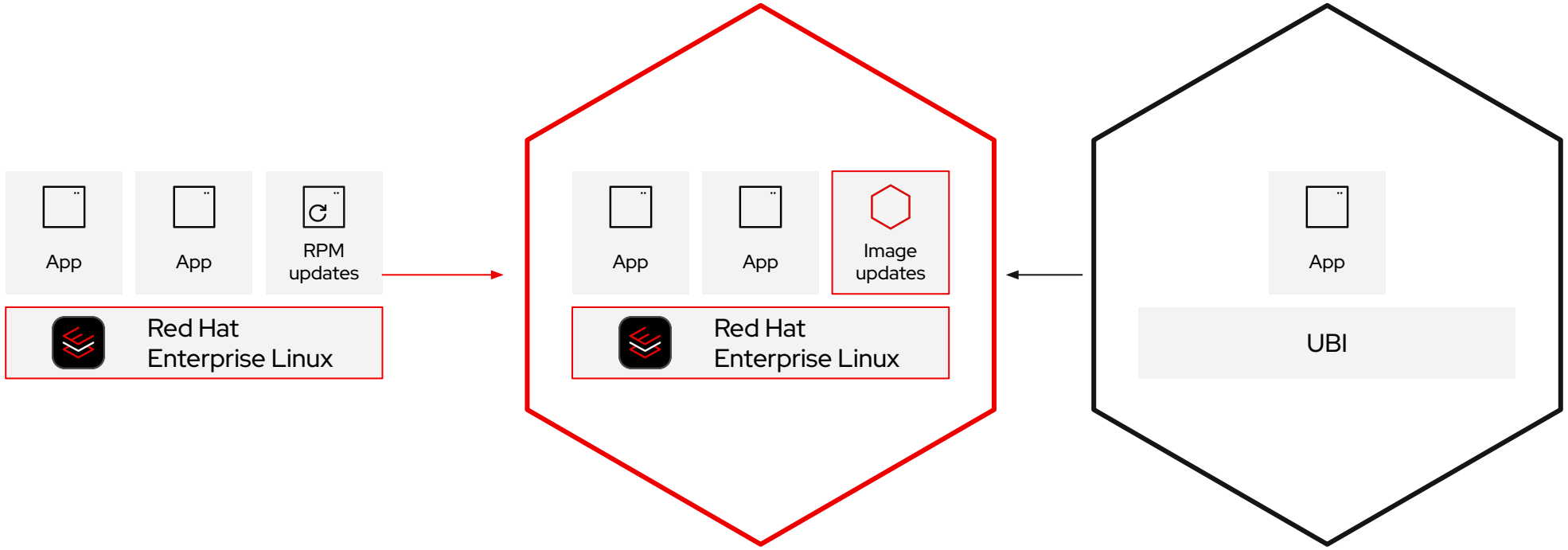
# Standardizing and innovating with containers




## Image mode for Red Hat Enterprise Linux is a simple, consistent approach to build, deploy and manage the operating system using container technologies.

Now you can manage the operating system with the same tools and workflows as applications, promoting a common experience and language across teams.

# Standardizing and innovating with containers




Package mode

Image mode

Application container



Red Hat

# Outcomes

What does image mode fix today?

## Less risk

Reduce the risk associated with updates with atomic transactions and rollbacks

## Better builds

Improve the composability and repeatability of standard builds through layering

## Move faster

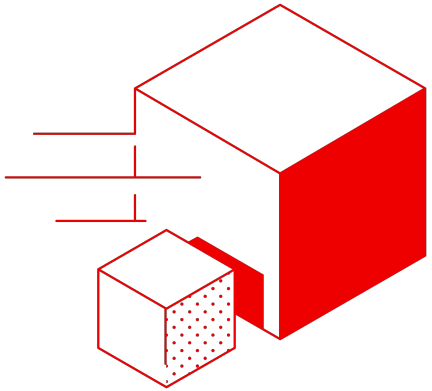
Increase the speed of experimentation

## Streamline process

Simplify end to end management with a single process for OS and applications

# Image mode for RHEL building blocks

- Dedicated bootc base image
- bootc CLI
- bootc-image-builder

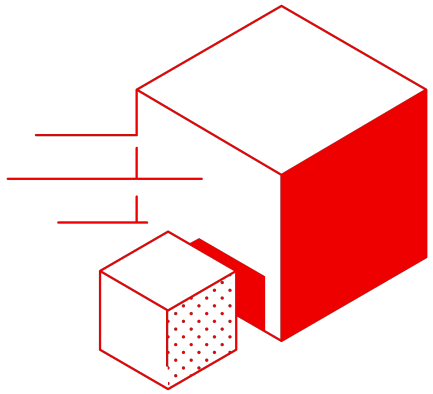




Red Hat

# Image mode for RHEL building blocks

- **Dedicated bootc base image**
- bootc CLI
- bootc-image-builder



A base image to rule them all!

Image Specs:

- 439 rpms
- ~785M compressed
- ~2.2G on disk

Primary contents:

- systemd, kernel, bootc
- rpm-ostree
- linux-firmware
- NetworkManager
- podman
- python
- Misc CLI tools: jq, sos

- Available on Quay.io:
  - registry.redhat.io/rhel10/rhel-bootc

# Image mode for RHEL building blocks

- Dedicated bootc base image
- **bootc CLI**
- bootc-image-builder

Manage the **bootc system** lifecycle

**`bootc upgrade`**
Download and stage an updated container image based on image reference.

**`bootc rollback`**
Rollback to the previous state.

**`bootc switch`**
Change to a different reference image

**`bootc install`**
Install container image **`to-disk`** or **`to-filesystem`**

- Upstream docs:
  - https://bootc-dev.github.io/

# Image mode for RHEL building blocks

- Dedicated bootc base image
- bootc CLI
- **bootc-image-builder**

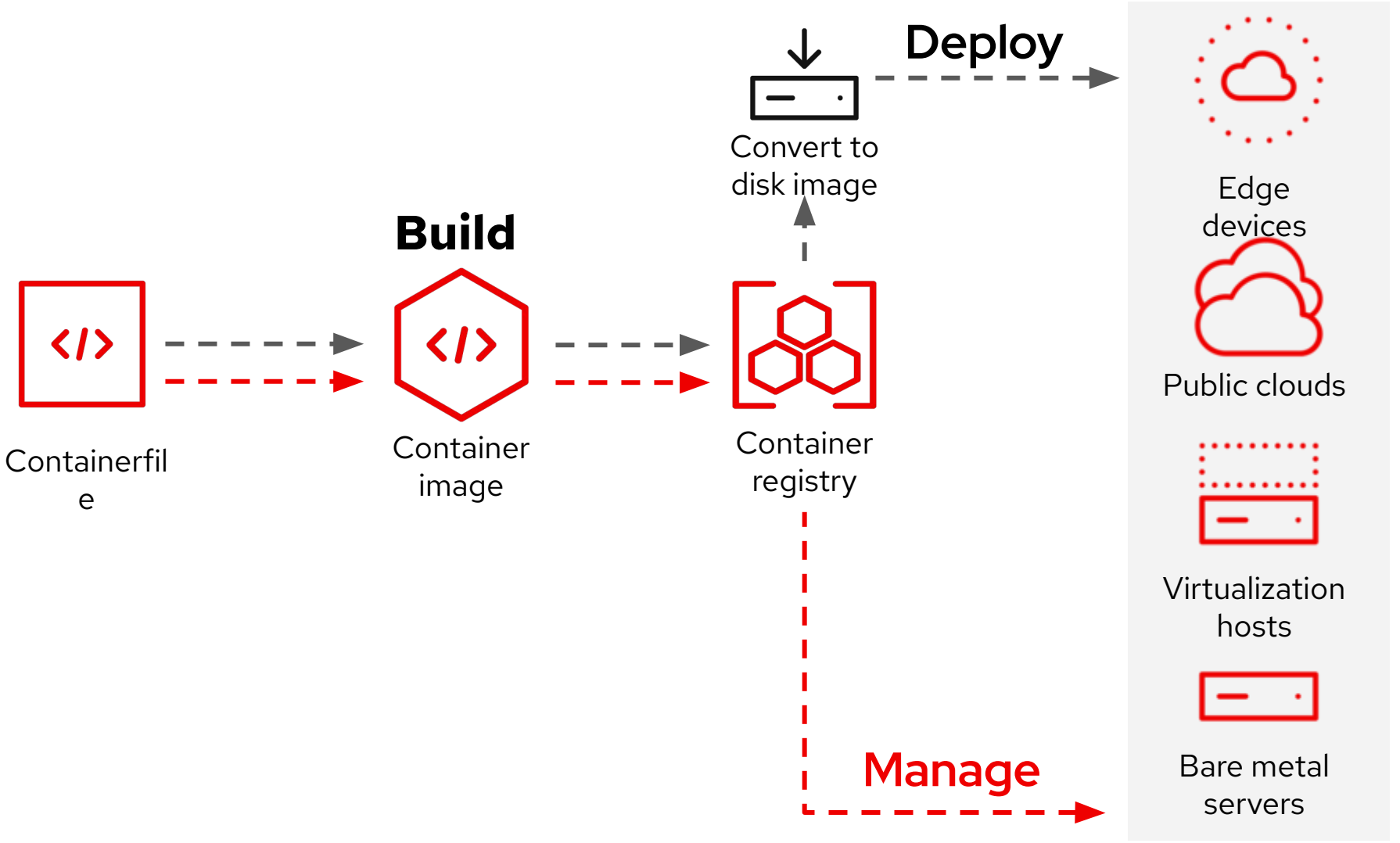
Convert **bootc images** to consumable formats

| | |
|---|---|
| `qcow2` | QEMU Disk Image |
| `ami` | Amazon Machine Images |
| `vmdk` | Virtual Machine Disk Image |
| `vhd` | Azure / Hyper-V disk image |
| `gce` | Google Compute Engine image |
| `raw` | MBR/GPT Raw disk image |
| `iso` | Bare Metal installer |

- Available as a container image:
  - registry.redhat.io/rhel10/bootc-image-builder
- Bootable Container Extension for **Podman Desktop**
  - Build for x86 & multi-arch builds

# Image mode for Red Hat Enterprise Linux

## Simple. Consistent. Anywhere.



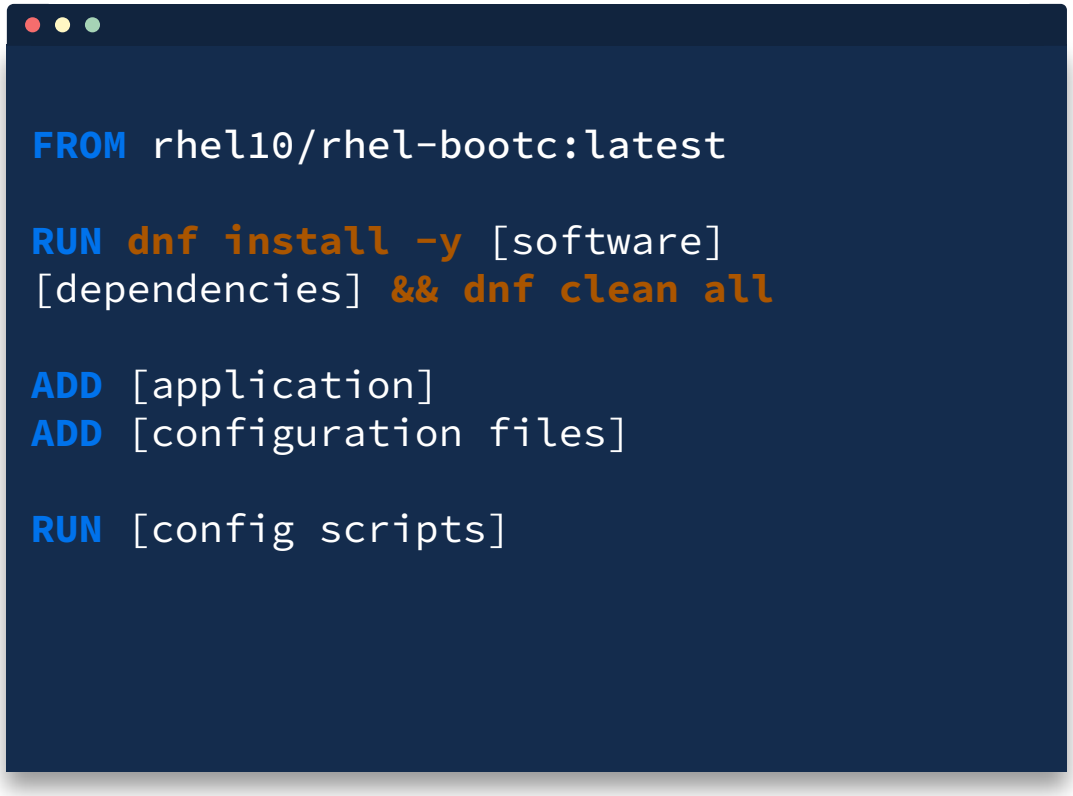



Red Hat

# Image mode for RHEL lifecycle

# Image mode for RHEL

A container-native workflow for the life cycle of a system




## Build

**A *bootc* base image & container file is all that's needed to describe a system, applications, and dependencies. Use your existing container tools or pipelines to quickly create and test images.**

## Deploy

Easily convert to a VM/cloud image or deploy on bare metal using RHEL. The container image includes full hardware drivers, but not cloud agents by default.

## Manage

Designed for modern GitOps & CI/CD driven environments. Systems will auto-update from the container registry by default. More advanced control and automation is available via custom rollouts (e.g. Ansible).

# Image mode for RHEL

Encapsulate differences in a sequence of builds

```
#Build standard operating environment
FROM rhel10/rhel-bootc:latest

RUN dnf install -y [system agents]
[dependencies] && dnf clean all

COPY [system config files]

RUN [config scripts]
```

# Image mode for RHEL

Encapsulate differences in a sequence of builds

```
#Build standard operating environment
FROM rhel10/rhel-bootc:latest

RUN dnf install -y [system agents]
[dependencies] && dnf clean all

COPY [system config files]

RUN [config scripts]
```

```
#Install App server on SOE
FROM corp-repo/corp-soe:latest

COPY [App]
COPY [App config files]
```

# Many ways to build

- Podman CLI
- Podman Desktop
- CI/CD and Automation Tools

podman

podman desktop

GitLab

GitHub

argo

Jenkins

TEKTON

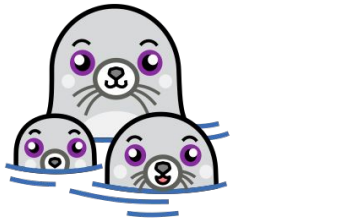
ANSIBLE



Red Hat

# Many ways to build

- **Podman CLI**
- Podman Desktop
- CI/CD and Automation Tools



podman

**Different content, same tools.**

- Build a bootc image like any other container image
- Integrate the image build phase in existing workflows and automation
- No need to reinvent the wheel, hassle-free implementation

# Many ways to build

- Podman CLI
- **Podman Desktop**
- CI/CD and Automation Tools


**An extension to rule them all!**

- Bootc Podman desktop extension available
- Build the Containerfile image directly from Podman Desktop
- Convert the image using the bootc-image-builder integration
- Additional extensions for: Kind, Kubernetes Contexts, MinC

# Many ways to build

- Podman CLI
- Podman Desktop
- **CI/CD and Automation Tools**

It works with your favourite tool!


- Example templates available!
  - https://gitlab.com/redhat/cop/rhel/rhel-image-mode-cicd



Red Hat

# Image mode for RHEL

A container-native workflow for the life cycle of a system

```
FROM rhel10/rhel-bootc:latest

RUN dnf install -y [software]
[dependencies] && dnf clean all

ADD [application]
ADD [configuration files]

RUN [config scripts]
```

## Build

A *bootc* base image & container file is all that's needed to describe a system, applications, and dependencies. Use your existing container tools or pipelines to quickly create and test images.

## Deploy

**Easily convert to a VM/cloud image or deploy on bare metal using RHEL. The container image includes full hardware drivers, but not cloud agents by default.**

## Manage

Designed for modern GitOps & CI/CD driven environments. Systems will auto-update from the container registry by default. More advanced control and automation is available via custom rollouts (e.g. Ansible).

# Install via Kickstart

Deploy container images to bare metal using installation media

```
lang en_US.UTF-8
keyboard us
timezone Etc/UTC --isUtc
text
zerombr
clearpart --all --initlabel
autopart
reboot
user --name=admin-user --groups=wheel
sshkey --username=admin-user "ssh-rsa
AAAAB3Nza......"

ostreecontainer --url quay.io/myimage:latest
```

NEW

## Use existing provisioning workflows

- RHEL/Fedora/CentOS boot media (isos)
- PXE & HTTP Boot for network based deployments

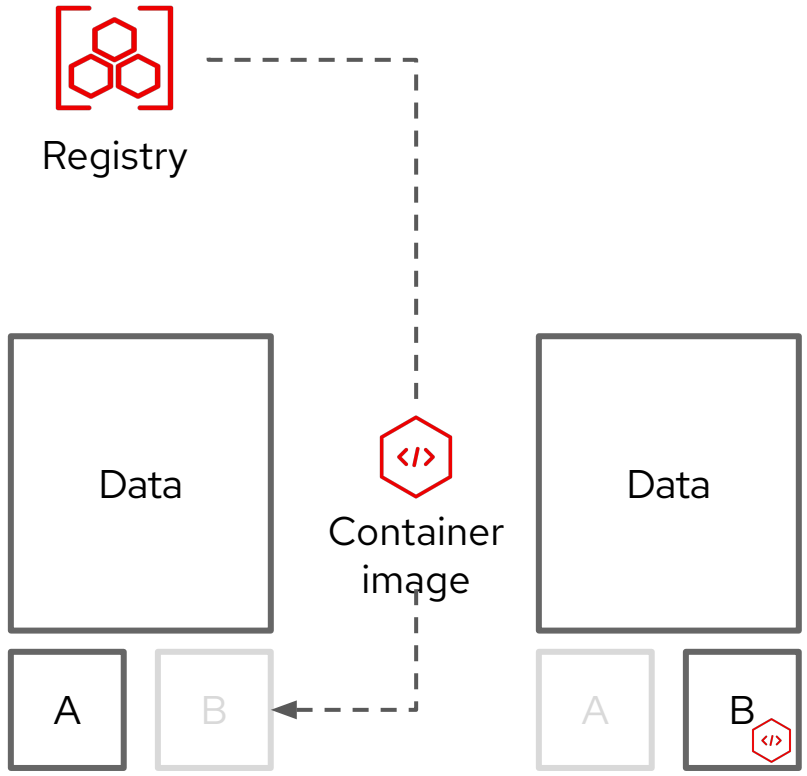
## Kickstart and Anaconda are used for disk layout and select configurations

- `%packages` is ignored
- ostreecontainer will fetch the container image from a registry and write it to disk.

## `%pre` and `%post` used for configuration

# Bootc: Image-based updates perfected

Immutable by default - secure by design




## Transactional updates (A → B model)

Bootc uses composefs and ostree to convert the container image into the root filesystem on the host..

## Roll forward or backwards

Updates are staged in the background and applied when the system reboots. The transactional model enables rollbacks for additional assurance

## Upgrades have never been easier

While there are some limits, bootc enables moving between minor releases (i.e. RHEL 9.4 → 9.5), as well as major releases (i.e. RHEL 9 → 10)



Red Hat

# Filesystem Layout

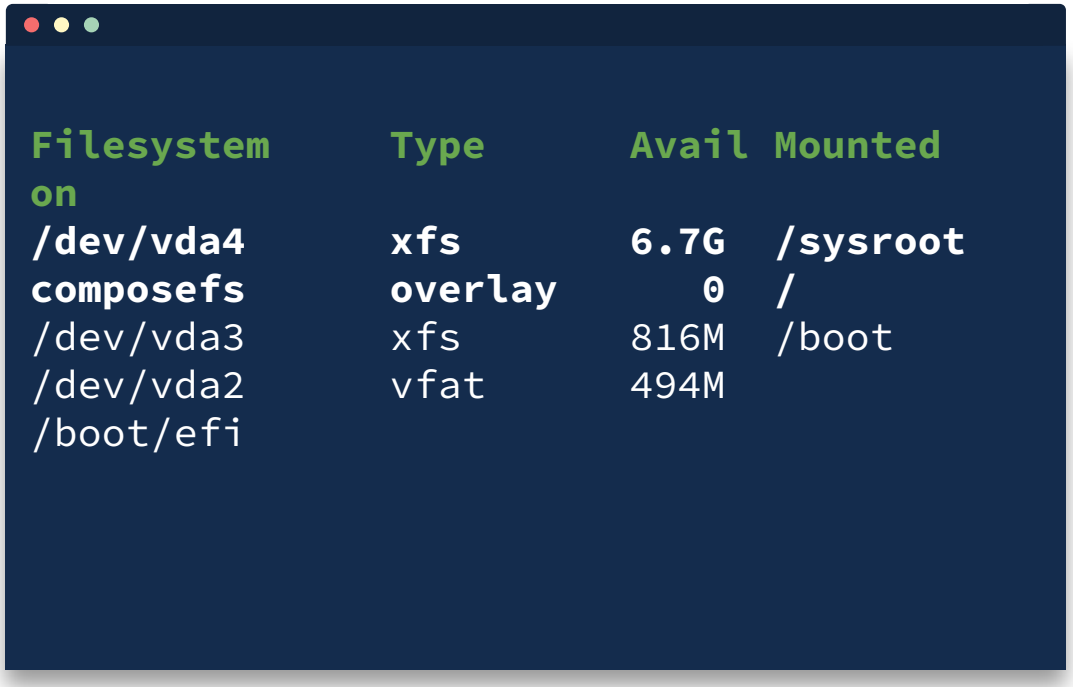
Similar to previous ostree setups - but better!!

**Build Time**
Everything is writable. e.g. /usr, /etc, /opt, ...

**Run Time**
All image content is read only

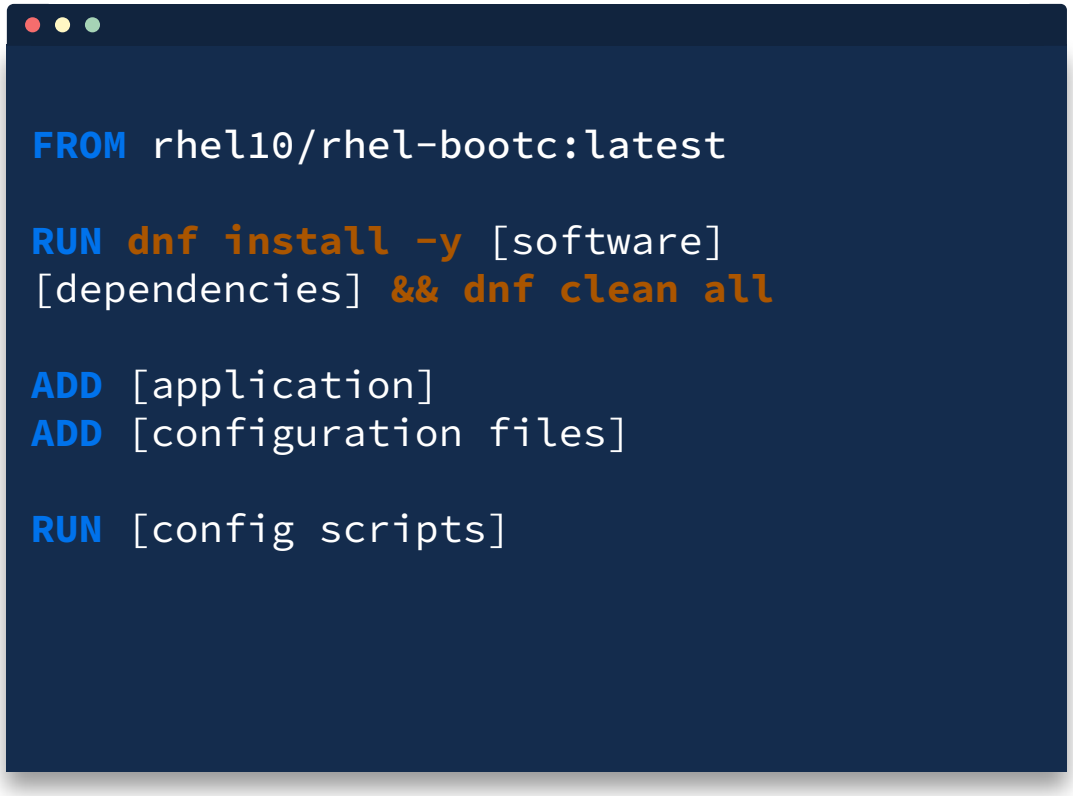
**/var** - RW, instance persistence. Not updated post install

**/etc** - RW, 3-way merge like Fedora/RHEL CoreOS. Machine local state (hostname, static IP)



# Image mode for RHEL

A container-native workflow for the life cycle of a system




## Build

A *bootc* base image & container file is all that's needed to describe a system, applications, and dependencies. Use your existing container tools or pipelines to quickly create and test images.

## Deploy

Easily convert to a VM/cloud image or deploy on bare metal using RHEL. The container image includes full hardware drivers, but not cloud agents by default.

## Manage

**Designed for modern GitOps & CI/CD driven environments. Systems will auto-update from the container registry by default. More advanced control and automation is available via custom rollouts (e.g. Ansible).**

# Management with Red Hat Insights

## Visibility and reporting made simple

### Inventory is easy

Image mode systems appear in inventory like package based systems

### Registration is simple

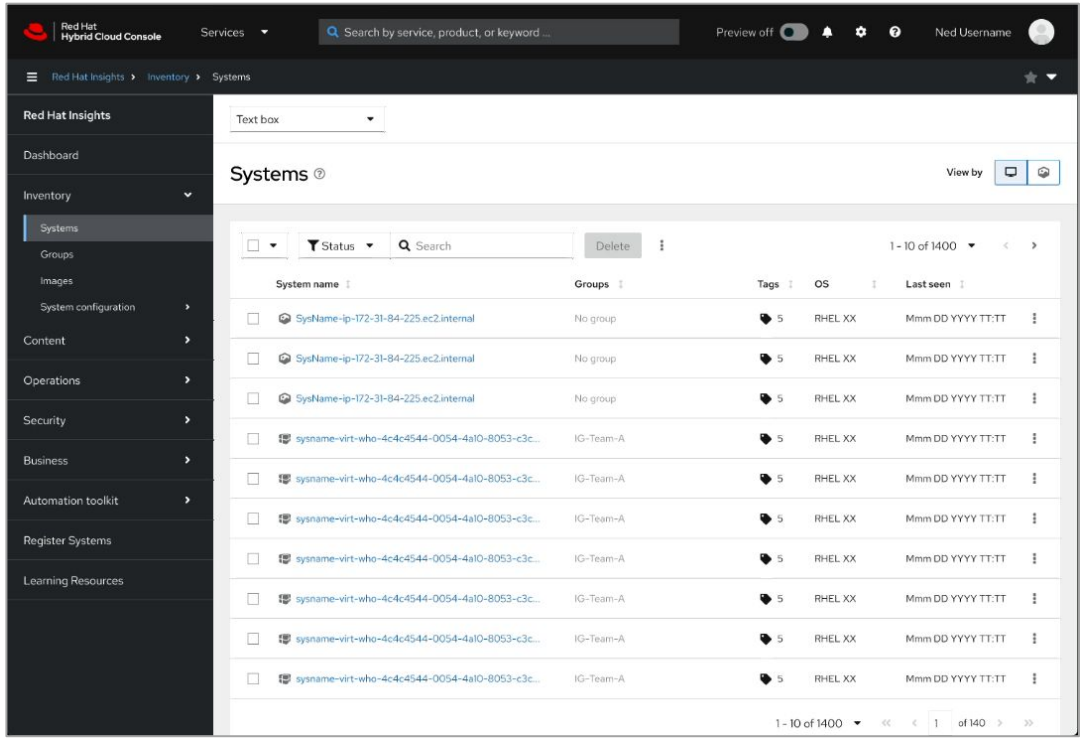
Activation keys can be baked into images via Containerfile, allowing auto registration to Insights at boot time

### Insights has insights

Image mode systems can be scanned for security and operational recommendations

### Updates are flexible

Image mode systems can be updated to new versions of images or remediated based on image-specific recommendations






Red Hat

# Red Hat Satellite 6.17

## Support for RHEL 10

**Inventory and manage** systems running Red Hat Enterprise Linux 10

## Image mode support

**New support** for provisioning, client management, and registry distribution for systems created with image mode for RHEL

## Support for Flatpak content

**Simplify** importing, managing, and deploying updates via containers with support for Flatpak content, including both custom packages and Red Hat-provided applications

## Secure boot support

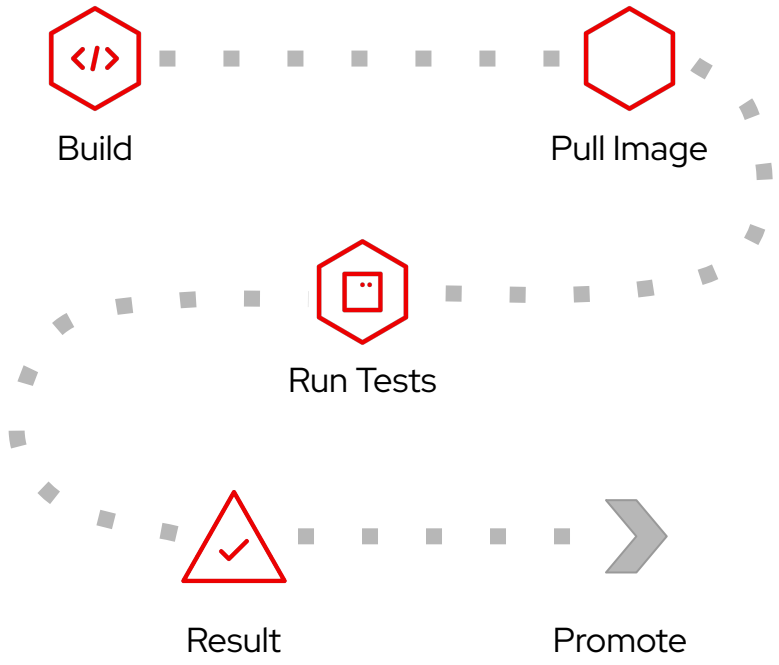
**Enhance** security during provisioning workflows on bare-metal, VMware vSphere, and Libvirt platforms

## IPv6 support

**Deploy** Satellite in an IPv6 environment, which helps address limitations of IPv4, including improved network efficiency and security

# Validating OS updates has never been easier

## CI pipelines used for apps now work with the OS




### Test/validate as a container

Bootc images are deployed as bare metal or VMs, but we can also run and test them **as containers**. This enables faster and lighter weight testing/validation of each build's userspace.

### Easy pipeline integration

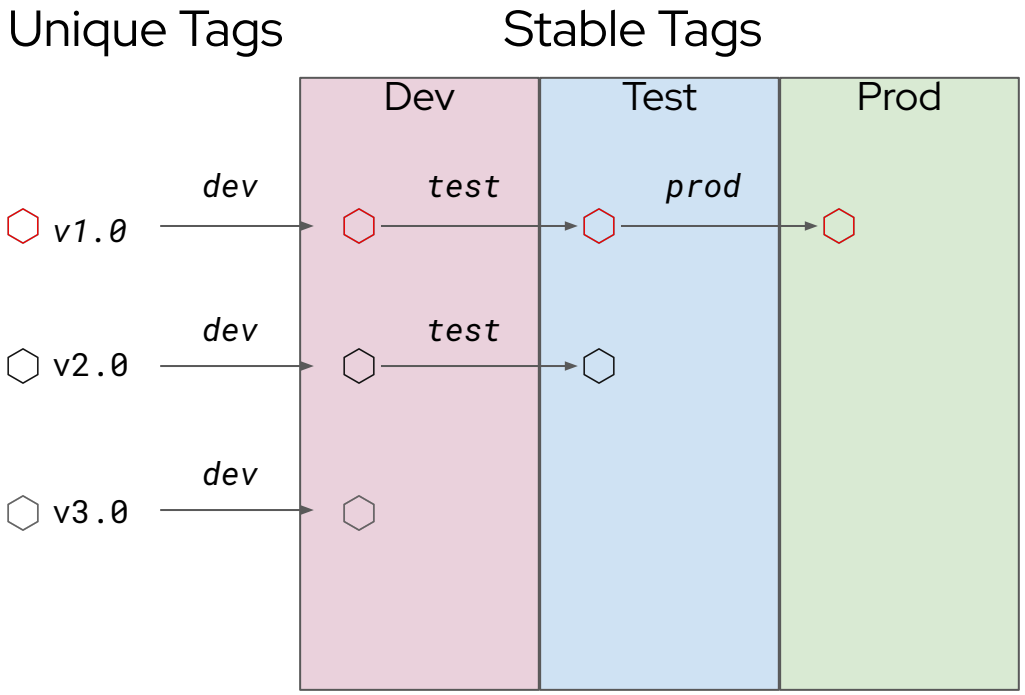
Containers have broad support across Github, Gitlab, Gitea, Circle CI, Jenkins, etc for the common container related tasks and testing. Use any system you like..

### Simple promotion through registry tagging

Tags are a powerful tool to identify dev → test → prod promotions.

# OS Updates via Container Registries

Tagging is powerful to version and promote updates




### Tags offer simple versioning and visibility

Tags are simple to automate and use for promotions. Bootc will default to updating from a `repository:tag`.

### Control updates via tagging

Combine tagging with the optional automatic updates to control fleets of systems via registry tags.

### Standardized & scaleable infra

Container registries scale very well and any standard registry can be used.

# Demo time!

Deploy an Image Mode based RHEL9 system on KVM and register it to Satellite

Upgrade to RHEL10 using Satellite

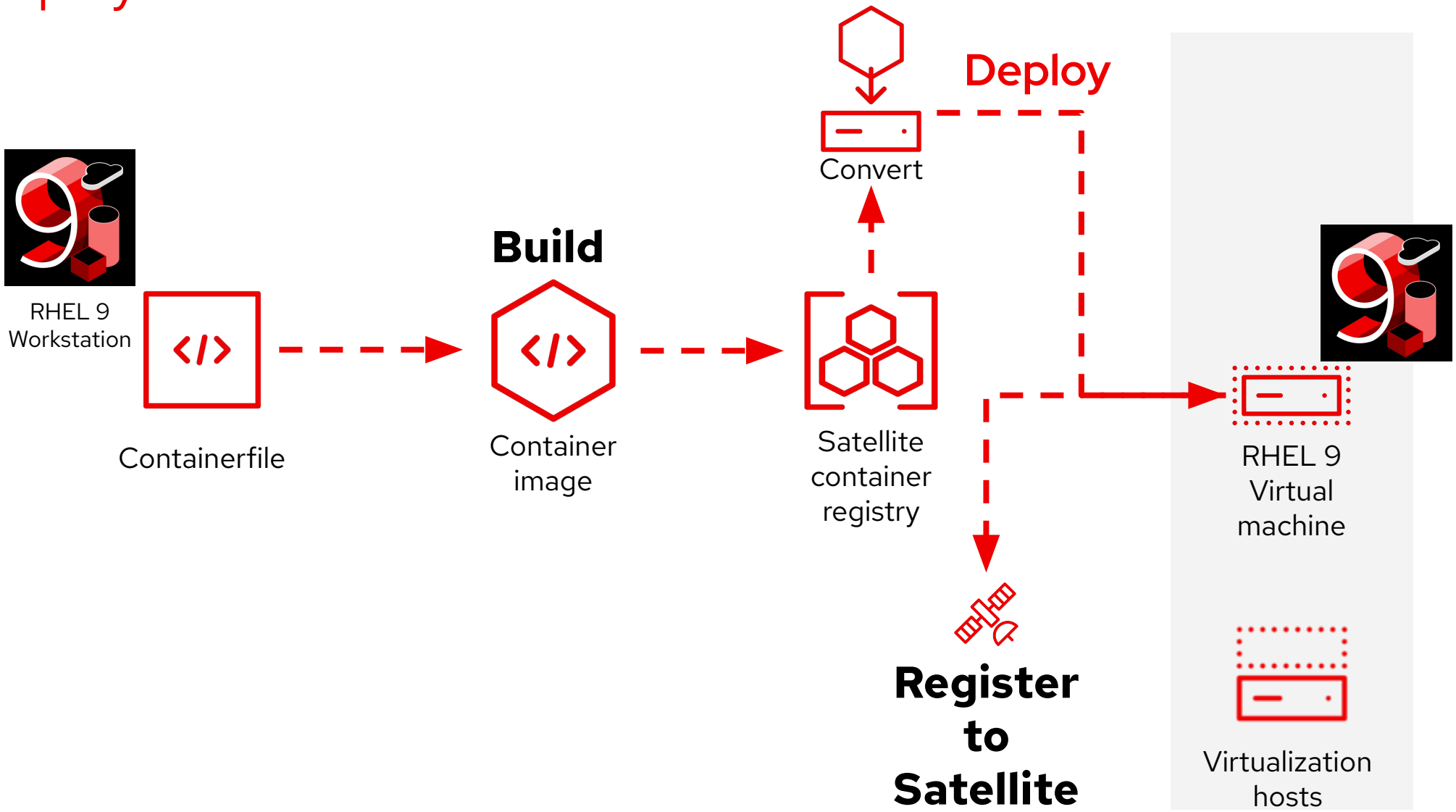
Red Hat

# Deploy an Image Mode based RHEL9 system on KVM and register it to Satellite

Red Hat

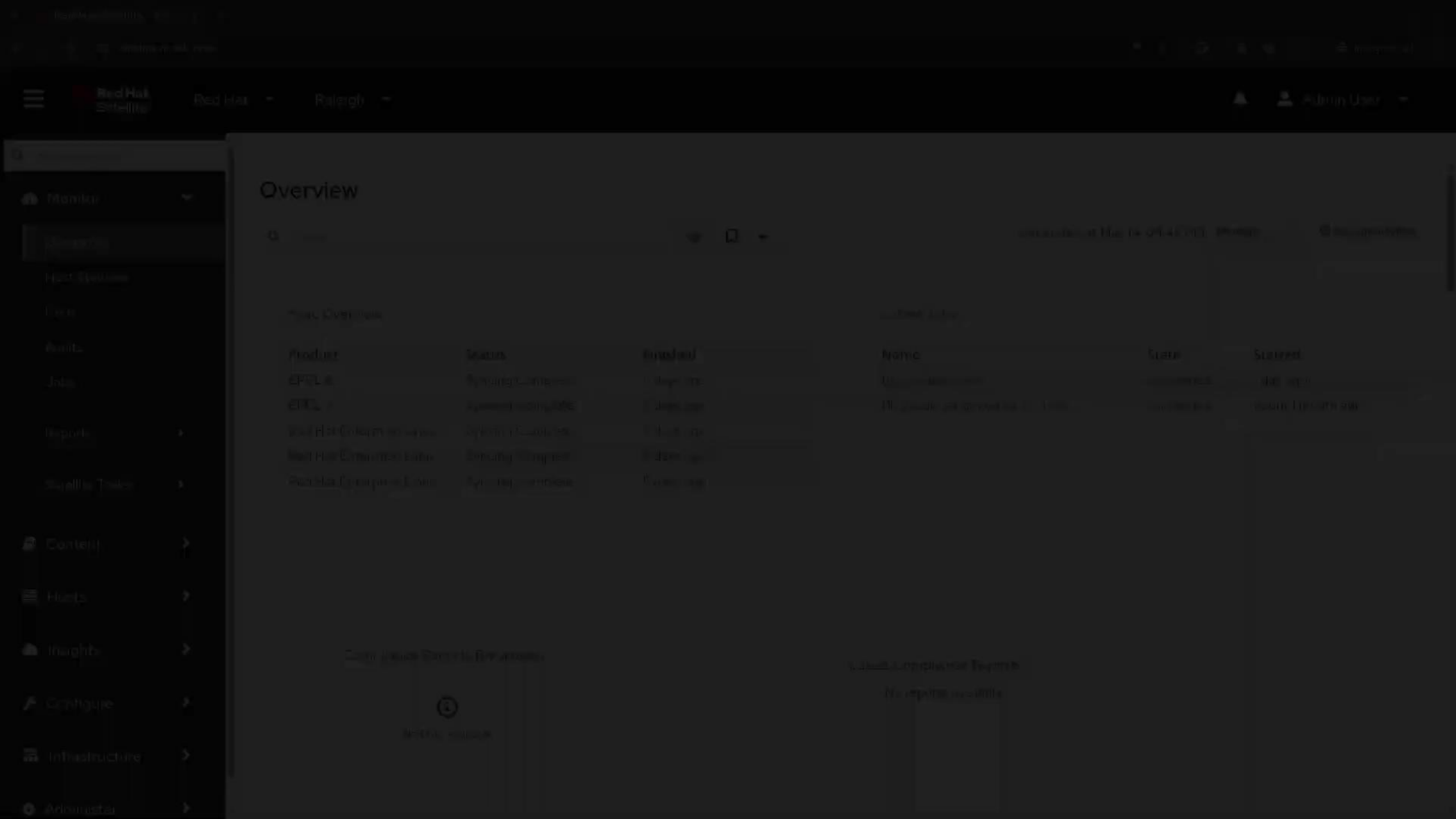
# Image mode Demo

## Deploy a RHEL 9 Workstation VM






Red Hat


Red Hat Satellite
Red Hat
Raleigh
Admin User
Overview
Monitor
Content
Hosts
Insights
Configure
Infrastructure
Administer

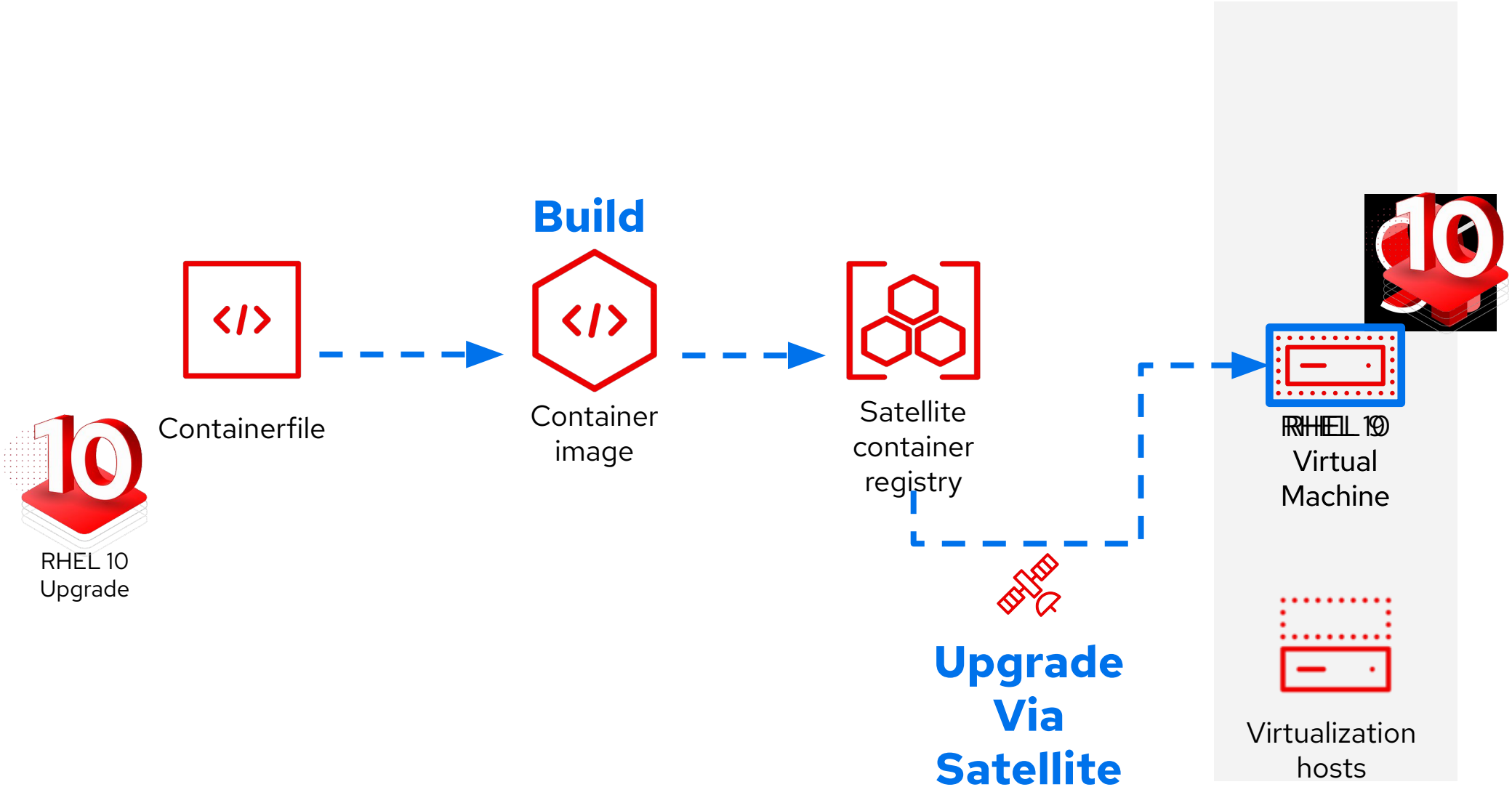
Deploy an Image Mode based RHEL9 system on KVM and register it to Satellite

Upgrade to RHEL10 using Satellite

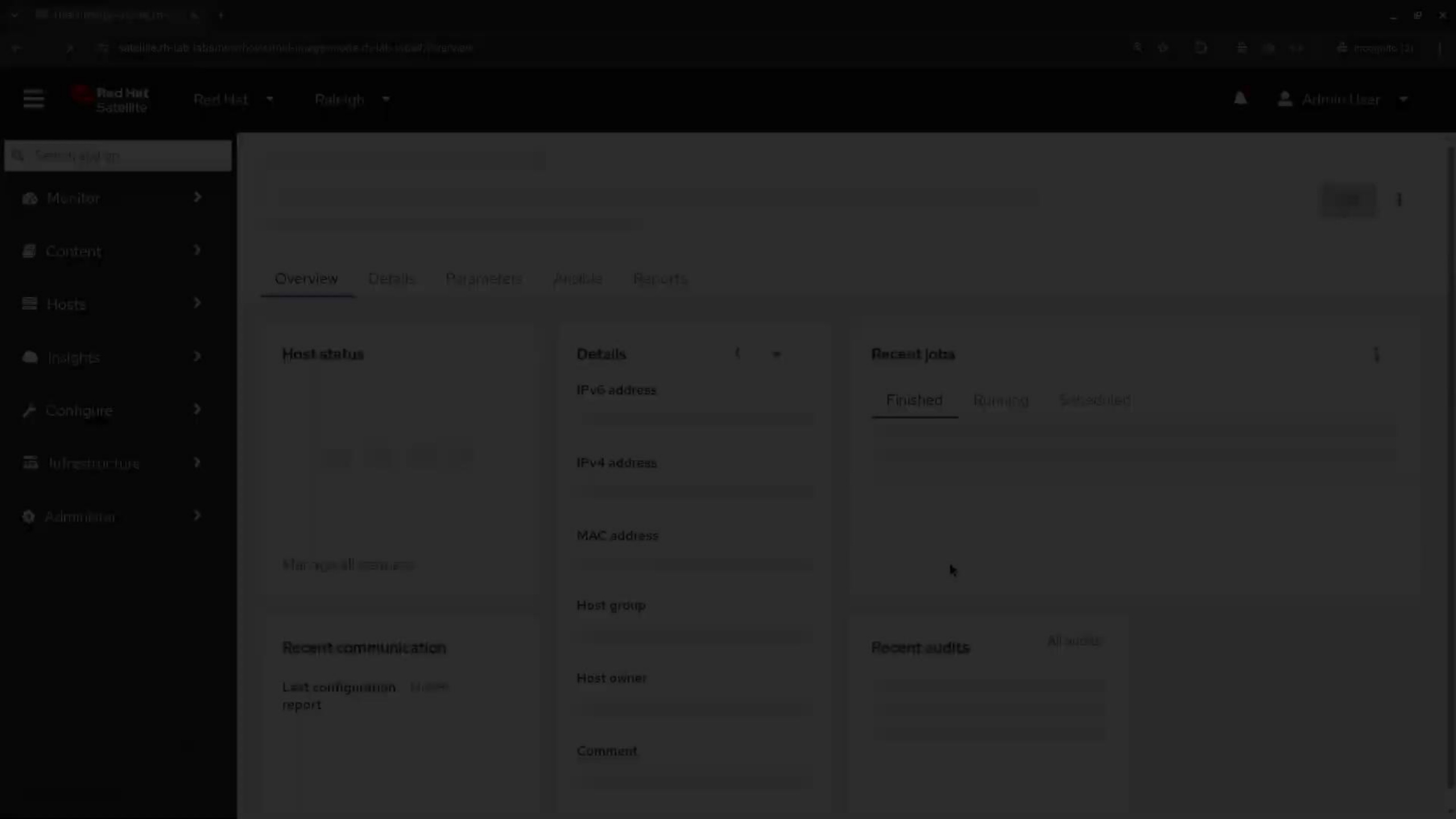
Red Hat

# Image mode Demo

## Upgrade the workstation to RHEL 10 using Satellite




Red Hat


Red Hat Satellite
Red Hat
Raleigh
Admin User
Monitor
Content
Hosts
Insights
Configure
Infrastructure
Administration
Overview
Details
Parameters
Ansible
Reports
Host status
Details
IPv6 address
IPv4 address
MAC address
Host group
Host owner
Comment
Recent jobs
Finished
Running
Scheduled
Recent communication
Last configuration report
Recent audits

# Use Cases and Adoption

# Use Cases

Where does image mode fit today?

Use Cases Blog

## AI/ML Stacks

Perfectly version app dependencies from kernel, GPU & accelerator drivers, frameworks, runtimes, etc

## 1:1 App/Host

Manage the OS AND app as a single unit

## Edge appliances

Easily manage a fleet of systems with registries and auto-updates

## Standalone container hosts

Use common toolchains and pipelines to build containerized applications and the hosting OS

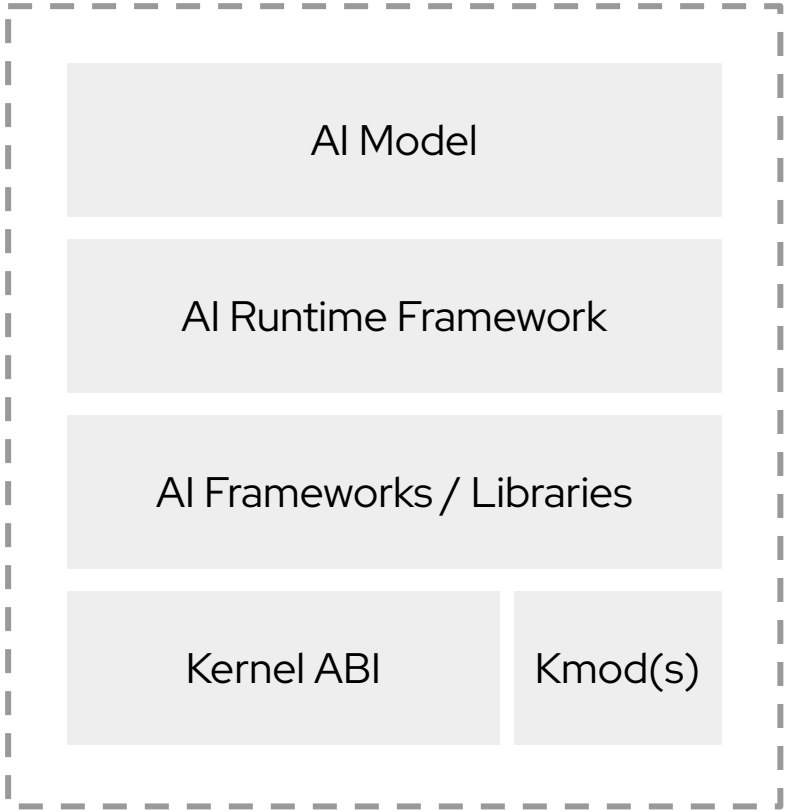


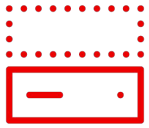
Red Hat

# AI/ML Stacks

## Deploy AI stacks confidently with image mode

- **Simple**: Much of the AI world already leverages containers, image mode helps deploy AI stacks quickly and efficiently.
- **Portable**: AI workloads often need to run in close proximity to data sources and image mode helps target multiple environments (better way to say that?)
- **Easy experimentation**: Image layering makes it effortless to test different models & frameworks and helps brings order to image sprawl.
- **Limit downtime**: version & test components and dependencies at build time before they go to production.

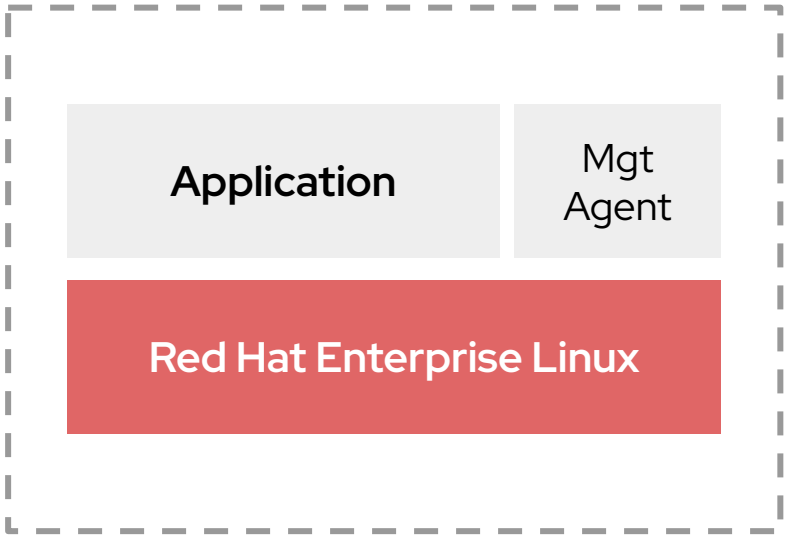




# 1:1 App/Host

Version & manage OS + app as a single unit

- **Efficient**: reduces the management footprint
- **Repeatable**: enforces consistency and reproducibility making it easy to “scale up”
- **Zero drift**: moves configuration to build time, leading to a more consistent fleet.
- **Rollbacks**: A/B boot model means fast recoveries in the case of unforeseen issues.
- Leverage containers for the *uncontainerizable* apps



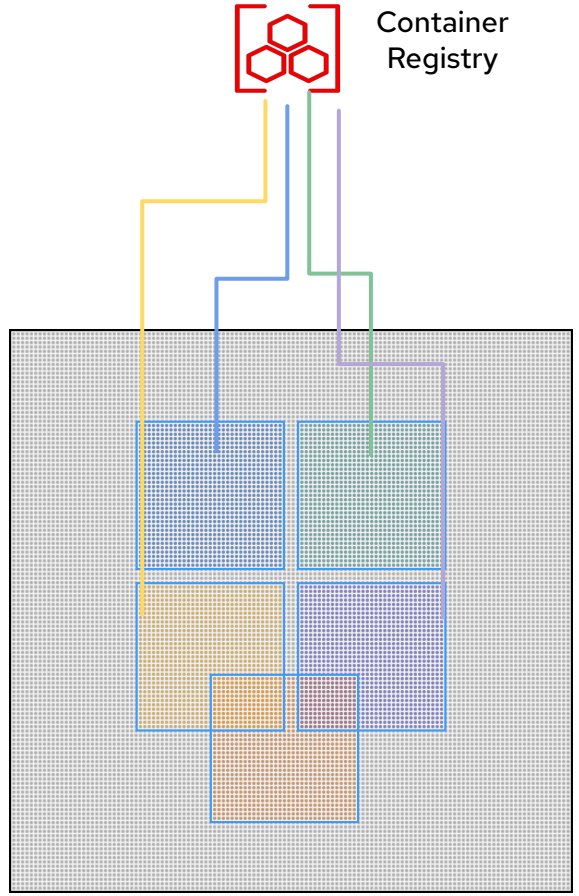



Red Hat

# Edge Appliances

Registries and auto-updates make managing a fleet of identical systems a snap

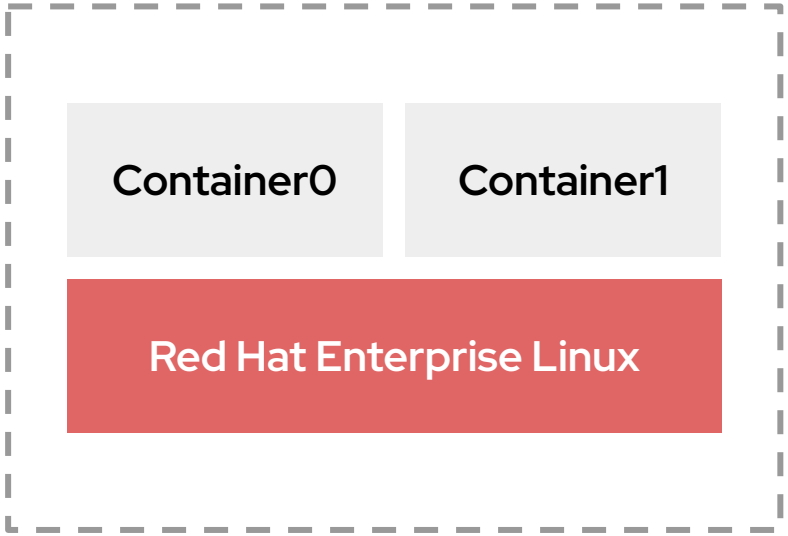
- **Image-based updates**: provide increased reliability over the life of the system.
- **Rollbacks**: A/B boot model means fast recoveries in the case of unforeseen issues.
- **Updates at scale**: Control OS & App container versioning through industry standard registries and  tags.
- Support for air-gapped and DIL environments



# Standalone Container Hosts

Simplify and manage the OS in the same way as your applications

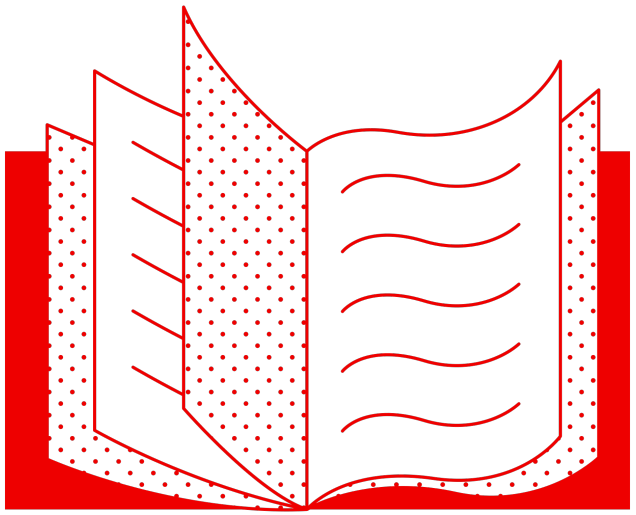
- **Flexible**: image mode provides a higher degree of host-level customization than previous immutable OSs.
- **Common tools**: use existing container tools and pipelines to build applications and the OS for better compatibility
- Align teams & operations around tools and process.



# Additional resources

# Useful resources

- RHEL Image mode on Red Hat Developers
  - https://developers.redhat.com/products/rhel-image-mode/overview
- RHEL Image mode documentation
  - https://docs.redhat.com/en/documentation/red_hat_enterprise_linux/9/html/using_image_mode_for_rhel_to_build_deploy_and_manage_operating_systems/index
- RHEL Image mode quickstart on Red Hat Blog
  - https://www.redhat.com/en/blog/image-mode-red-hat-enterprise-linux-quick-start-guide
- RHEL Image mode overview - YouTube
  - https://www.youtube.com/watch?v=QZDaTHyl1Sk
- RHEL Image mode Lab fast forward instructions
  - https://github.com/AutomationWitch/instruqt-mods/tree/main/image-mode



# Available Labs

## Image Mode Basics

Build and deploy an Image Mode system - https://red.ht/lab-image-mode

## Image Mode Day 2 Operations

Manage a virtual machine running in Image Mode - https://red.ht/lab-image-mode-day2

## Red Hat Satellite Basics

Basics of what is new in Satellite 6.17 - https://red.ht/lab-satellite-basics

## Satellite Advanced Topics 6.17

Advanced lab on what is new in Satellite 6.17 - https://red.ht/lab-satellite-advanced

**https://labs.redhat.com**

# Try it yourself!

The whole demo shown in this session and other use cases are available in the following Github repository:

https://red.ht/rhel-image-mode-demo

Everybody is welcome to use it, fork and suggest improvements.

Red Hat Summit

Connect

# Thank you


linkedin.com/company/red-hat


facebook.com/redhatinc


youtube.com/user/RedHatVideos


twitter.com/RedHat