RAPIDFORT

WHITEPAPER

Your Path to Near-Zero CVE Images

A practical approach to choosing secure, well-maintained container images and the vendors behind them.

CONTAINER SECURITY | SUPPLY CHAIN | DEVSECOPS



© 2026 RapidFort, Inc. | www.rapidfort.com

01 Executive Summary

Securing Software Earlier in the Pipeline

In today's cybersecurity landscape, the focus has shifted left, emphasizing the importance of securing software earlier in the CI/CD pipeline rather than waiting until after deployment. One critical strategy for achieving this is using near-zero CVE operating system, framework, and third-party application images. These are container images that are regularly patched and hardened to eliminate known Common Vulnerabilities and Exposures (CVEs), making them essential building blocks for modern containerized software development.

This guide explains how to evaluate near-zero CVE images and the factors to consider when selecting a vendor.

85%
of enterprise software is open source

235,000+
CVEs with known fixes identified, 2016 to 2025

10 hrs
from vulnerability disclosure to active exploit

Most enterprise risk is inherited from the open source components an application is built on, not written by the development team. Choosing the right base image, and the right vendor behind it, is one of the highest-leverage decisions a team can make to reduce that inherited risk.

02 Selection Criteria

Start with a Mature and Widely Supported Operating System

Start by selecting a vendor that curates images on a well-supported, well-maintained, and widely adopted OS distribution. Choose a distribution with an active and broad community that regularly identifies vulnerabilities, contributes patches, and provides long-term support, CIS benchmarking, and other security and stability measures.

Ubuntu, Debian, Red Hat, and Alpine each have a significant, well-established presence in the container ecosystem. Ubuntu provides a familiar, widely adopted base. Alpine offers a lightweight option suited to minimal images. Debian and Red Hat's Universal Base Image (UBI) are backed by mature, long-standing communities with established support models.

Ubuntu

Widely adopted LTS distribution known for stability, broad package support, and an active community that drives continuous testing and rapid vulnerability detection.

Debian

A long-standing, community-governed distribution with a strong reputation for stability and transparent security practices.

Red Hat (UBI)

An enterprise-grade base image backed by a mature vendor and a broad contributor community, widely used in regulated environments.

Alpine

A minimal, lightweight distribution well suited to reducing image size and attack surface in container workloads.

⚠

Avoid operating systems maintained by a single company without broader community or market adoption. A distribution with no independent contributor base creates vendor lock-in and increases the risk that, if the vendor experiences business difficulties, the OS and its unique packages are left unsupported.

03 Image Strategy

Selecting Scratch vs. Curated Base Images

Scratch Images

Scratch images are suitable for greenfield projects, where a small team working closely together builds an image from nothing and adds only the specific packages their code needs. This approach can produce leaner, more minimal images. In practice, though, it requires continuous updates as dependencies change with every code drop, and the likelihood of breaking the image increases with every iteration.

Security Technical Implementation Guides, such as the CIS Benchmark or DISA STIG, include hundreds of configuration recommendations, so hardening and auditing a Linux system manually can be tedious. Tools such as the Ubuntu Security Guide improve the usability of hardening and auditing and allow for environment-specific customization.

Most teams are working in brownfield environments: legacy codebases, or applications originally built by developers who have since moved on. Understanding the precise dependencies in these environments can be difficult and time-consuming.

Curated Base Images

In these cases, it is often better to start with curated base images that already include the necessary packages and have already been patched. This reduces the risk of runtime breakage caused by missing dependencies. Teams can still optimize these images further using tools that harden them based on runtime profiles, which produce more reliable results than manual dependency tracking.

Profiling a container's runtime behavior across multiple instances with every code iteration helps maintain an accurate dependency tree and ensures the workload stays hardened for subsequent builds.

04 Vendor Evaluation

Choose an OS Distribution with Trustworthy Security Advisories

When selecting an OS distribution, it is important to choose one with a transparent and reliable security advisory process. Distributions that benefit from large, diverse user communities, where a broad base of contributors reports vulnerabilities and reviews patches through mature QA cycles, support a more accurate and transparent assessment of security risk.

By contrast, a distribution managed and audited entirely by a single company presents a conflict of interest. When the same organization that builds a distribution is also solely responsible for reporting its vulnerabilities, objectivity can be difficult to maintain. A large, active, third-party community brings independent scrutiny to security and CVE reporting, helping ensure that risks are surfaced rather than managed quietly behind closed doors.

Use widely supported distributions recognized by mainstream scanners

Image scanning tools are only as capable as the components behind them: a scanner to detect software components, a database to evaluate them against, and a reporting layer to present the results. Scanners may not detect every third-party component in a codebase, and vulnerability databases may not document libraries from small vendors or lesser-known open source projects.

If a "near-zero CVE" image vendor relies on an obscure or custom distribution, mainstream scanners may fail to recognize the container's origin and skip scanning it altogether. This gap makes it difficult to validate, reconcile, and trust a vendor's near-zero CVE claim. A distribution that mainstream scanners cannot properly assess can create a false sense of security. To support comprehensive security assessments, choose distributions that are widely supported and easily identifiable by industry-standard scanning tools.

Community Trust

Broad contributor base ensures objective vulnerability reporting and transparent patch cycles.

Scanner Visibility

Widely adopted distros are recognized by Trivy and Wiz, enabling full assessments.

No Hidden Risk

Mainstream distributions surface risk openly rather than managing it behind closed doors.

05 Final Thoughts

Building a Secure Foundation

Adopting near-zero CVE images is a critical step toward building a resilient software supply chain. Starting with a mature and widely supported OS distribution, using well-curated images to manage dependencies, and choosing vendors with transparent CVE advisories and broad scanner support are the key steps to building a secure software foundation.

Selecting near-zero CVE images that allow flexibility to add necessary packages also ensures that software remains adaptable as requirements change. Following these principles allows teams to build modern, containerized applications that are secure and efficient from the ground up, with vulnerabilities eliminated at the source rather than managed after the fact.

- Start with a mature, widely supported OS distribution
- Use well-curated images to manage dependencies predictably
- Choose vendors with transparent CVE advisories
- Ensure broad scanner support for independent validation
- Maintain flexibility to add packages as requirements evolve

WHERE RAPIDFORT FITS

RapidFort Curated Images deliver tens of thousands of or more near-zero CVE images across Ubuntu, Debian, Red Hat, and Alpine, hardened to NIST 800-70 and FIPS 140-3 validated, and benchmarked against STIG and CIS. Images are drop-in and pin-for-pin compatible, with no vendor lock-in and no proprietary distribution to depend on. This lets teams follow the guidance in this paper without trading one form of risk for another.

06 The RapidFort Platform

Three Steps to Continuous CVE Elimination

RapidFort is the leader in Software Supply Chain Security, eliminating risk across the software stack at scale through a single platform spanning intake, build, and runtime.

STEP 01

Intake

RapidFort Curated Images

- Near-zero CVE images, hardened to NIST 800-70
- FIPS 140-3 validated
- STIG and CIS benchmarked
- Continuously patched

STEP 02

Build

RapidFort Analyzer and Optimizer

- Generates an SBOM and identifies real CVE exposure
- Removes unused components to reduce attack surface

STEP 03

Run

RapidFort Profiler, Runtime, and Continuous Compliance

- Continuously scans for malware & tampering
- Separates real, exploitable risk from theoretical CVE noise
- Continuously monitors and protects deployed environments
- Compiles audit-ready compliance evidence

Together, these steps eliminate up to 99.9% of vulnerabilities within hours and reduce the attack surface by up to 90%, without code changes.

Up to 99.9%
Vulnerabilities Eliminated

Up to 90%
Attack Surface Reduction

RAPIDFORT

ABOUT RAPIDFORT

RapidFort is the leader in Software Supply Chain Security, enabling organizations to eliminate risk across their software stack at scale. Its platform combines curated near-zero CVE container images, runtime profiling, and attack surface management to remove up to 99.9% of vulnerabilities within hours and reduce the attack surface by up to 90%, without code changes. RapidFort appears in the inaugural Gartner® Magic Quadrant™ for Software Supply Chain Security and is a Gartner® Cool Vendor™ and Nutanix Global Cloud Native Partner of the Year.

© 2026 RapidFort, Inc.

Eliminate attack vectors at the source

Schedule a Call ↗

www.rapidfort.com | sales@rapidfort.com