be//soft

THE 2025 STATE OF CONTAINER SECURITY

EXECUTIVE SUMMARY

The container ecosystem has matured significantly over the past decade, yet fundamental questions about security practices remain unresolved.

To better understand how teams manage container security today, we surveyed 427 professionals at Devoxx 2025 to examine:

- how organizations select and build container images,
- which security practices they follow,
- what challenges they encounter, and
- where current practices fall short of their stated priorities.

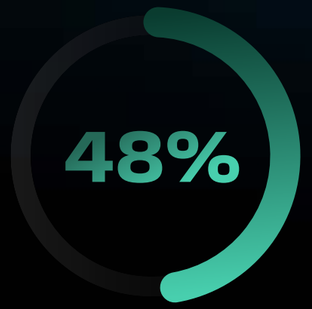
Across the responses, a clear pattern emerges: **teams overwhelmingly value security, efficiency, and simplicity — yet their tools, OS choices, and JVMs directly undermine these goals.**

KEY FINDINGS

experienced container-related security incidents in the past year

cite time and resource constraints as a primary challenge in maintaining container security

update monthly or less frequently. Meanwhile, exploitation windows have shrunk from weeks to days.



want pre-hardened base images, indicating a shift toward outsourcing vulnerability management.

TEAMS KNOW WHAT GOOD SECURITY LOOKS LIKE, BUT CURRENT TOOLING, OS CHOICES, AND JDKS MAKE IT DIFFICULT TO ACHIEVE WITHOUT EXTERNAL SUPPORT.

29% of respondents prioritize security (minimal number of CVEs) when selecting base container images.

2/3 of respondents want efficiency for Java applications in containers, measured by low memory usage, high throughput, and fast startup time.

VS

55% use general-purpose Linux distributions with hundreds of unnecessary packages and unresolved CVEs.

69% use general-purpose JDKs requiring manual optimization, consuming engineering time and inflating cloud bills.

SECURITY LEADS, BUT PRIORITIES PULL IN DIFFERENT DIRECTIONS

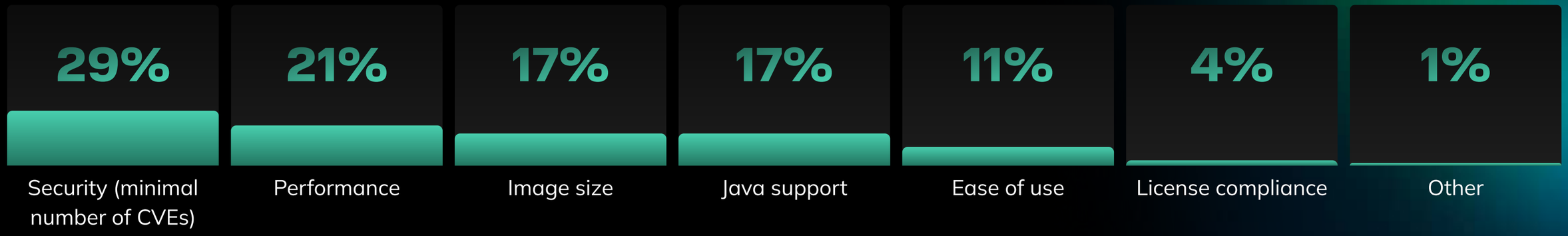
Security dominates at 29%, followed by performance at 21%, image size at 17%, and Java support at 17%. This split reveals the challenge: teams manage multiple concerns simultaneously.

Image size and memory efficiency directly impact cloud spending. Smaller containers mean lower storage costs, faster deployments, and reduced bandwidth. With hundreds or thousands of instances running, these differences compound quickly.

Java applications complicate this further. General-purpose JDKs weren't built for containers. Organizations either invest significant engineering time in manual tuning or accept higher cloud bills.

Solutions like Liberica JDK Lite deliver 30% reduction in memory and disk space without complex configuration.

WHAT IS THE MOST IMPORTANT FACTOR FOR YOU WHEN CHOOSING A BASE CONTAINER IMAGE?



PERFORMANCE OPTIMIZATION REQUIRES EXPERTISE MOST TEAMS LACK

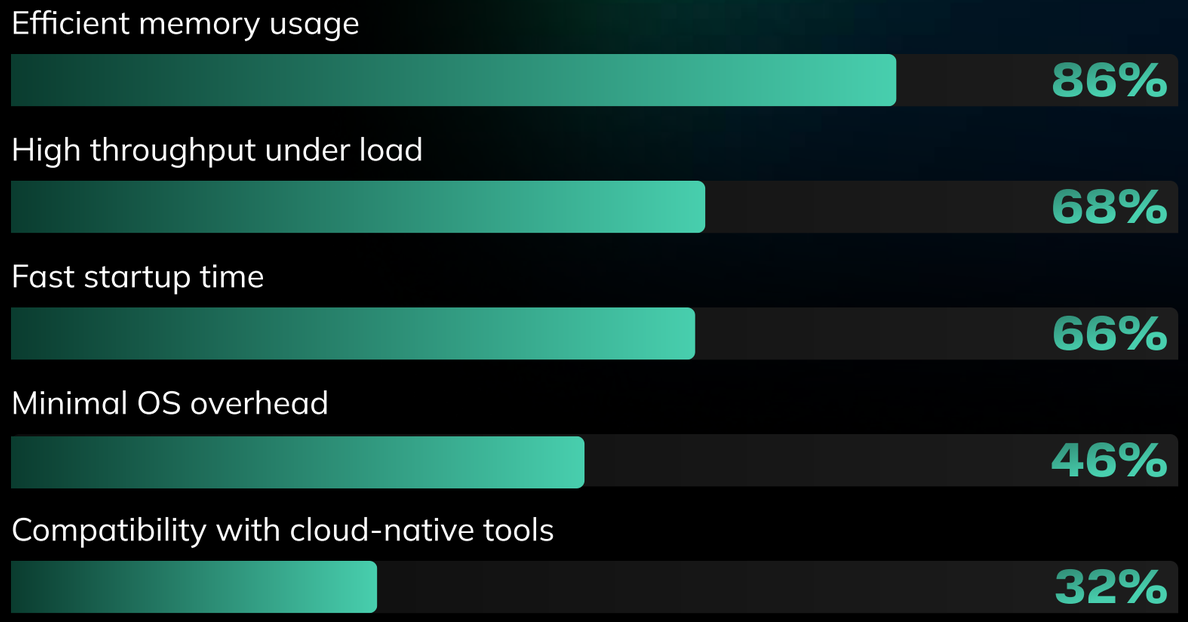
RESPONDENTS OVERWHELMINGLY PRIORITIZE:

1. Efficient memory usage (86%)
2. High throughput (68%)
3. Fast startup (66%)

But achieving these requires deep knowledge of container-aware GC tuning, heap sizing, CDS configuration, or advanced approaches like GraalVM and CRaC—complex and not always applicable.

Out-of-the-box optimized runtimes, such as lightweight Java distributions, eliminate this effort and deliver consistent performance gains without extra engineering overhead.

WHAT PARAMETERS ARE CRITICAL FOR YOUR COMPANY FOR JAVA APPLICATION PERFORMANCE IN CONTAINERS?



OPERATIONAL CONVENIENCE VS SECURITY BEST PRACTICES

TEAMS RELY ON TOOLS LIKE:

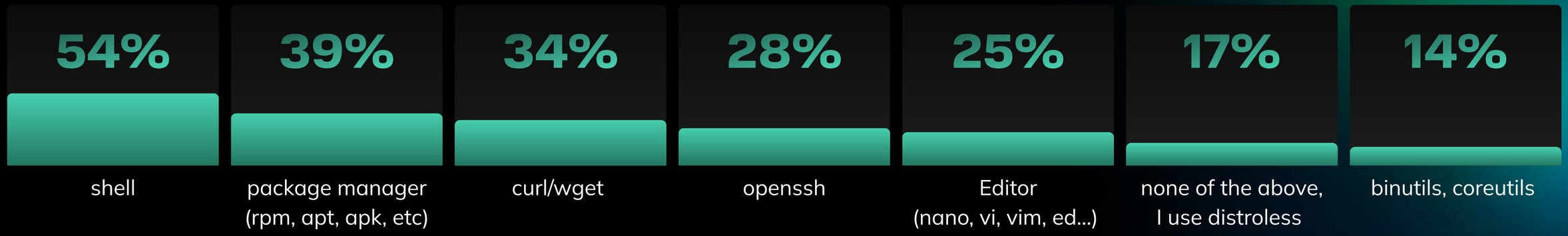
- Shells (54%)
- Package managers (39%)
- curl/wget (34%)

These are helpful for debugging but significantly expand the attack surface. Only 17% use minimal/distroless images in production.

Organizations face a real conflict: **operational convenience during development vs. minimal attack surface in production.**

A practical approach is using hardened minimal runtime images, paired with fuller “debug builds” during development—allowing both security and diagnostics without compromise.

WHICH OF THE FOLLOWING DO YOU FIND ESSENTIAL INSIDE THE BASE CONTAINER?



HALF CHOOSE BLOATED DISTRIBUTIONS, HALF FACE SUPPORT GAPS

The split is revealing: 41% use Alpine while 55% use Ubuntu/Debian or Red Hat-based systems. This represents fundamentally different approaches.

Ubuntu/Debian and Red Hat users rely on general-purpose distributions with hundreds of packages their applications never use. Each represents potential vulnerabilities requiring security patches. When a vulnerability emerges, security teams must evaluate impact and coordinate updates across thousands of instances—regardless of whether the application uses the affected package.

Alpine Linux users chose minimal distributions built for containers, reducing image size and vulnerability exposure. However, Alpine is community-driven with no commercial support, no SLAs, and no Long-Term Support releases. For enterprises under regulatory frameworks, this creates compliance concerns.

Alpaquita Linux addresses this gap—delivering minimal attack surface with enterprise-grade support guarantees.

BASE OPERATING SYSTEMS:

- **41%** Alpine Linux
- **34%** Ubuntu/Debian
- **21%** Red Hat/Red Hat-based distributions (Rocky, AlmaLinux, CentOS)
- **2%** Alpaquita Linux
- **1%** other distributions

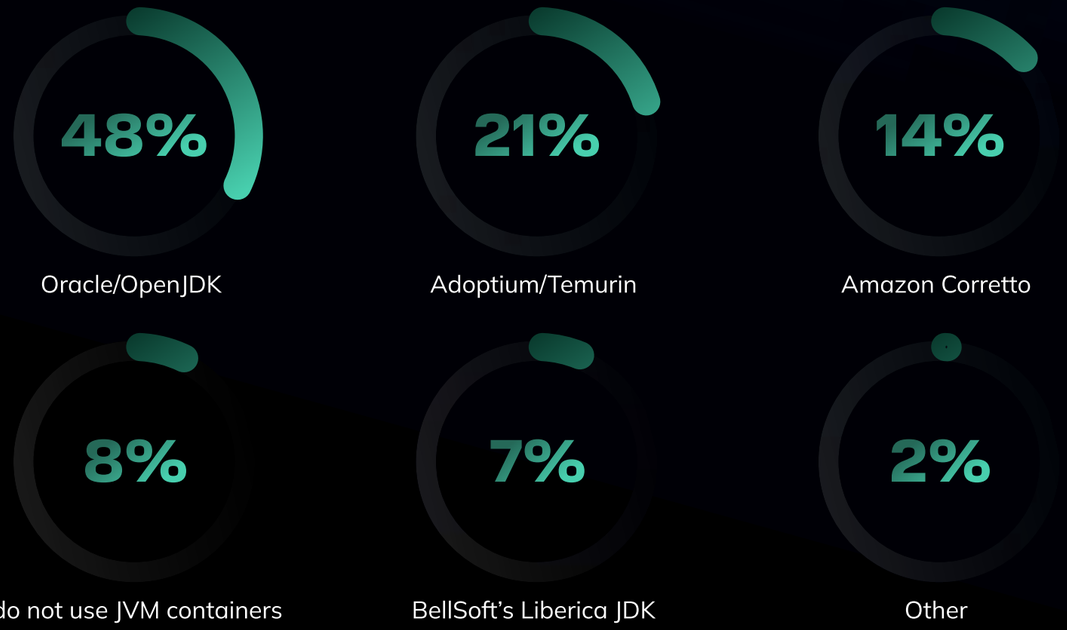
MAINSTREAM JDKS CARRY HIDDEN SECURITY & COST BURDENS

Over 69% use Oracle/OpenJDK or Adoptium/Temurin—mainstream distributions aligned with what developers know. This familiarity has hidden costs.

These general-purpose JDKs weren't optimized for containers. They consume more memory, produce larger images, and require substantial tuning for acceptable performance. Standard distributions regularly ship with known CVEs requiring continuous tracking and patching. Oracle's commercial JDK adds licensing complexity that changes unpredictably.

Alternatives designed for containerized environments, like Liberica JDK Lite. It delivers optimized performance without specialized tuning, maintains a minimal vulnerability profile, and provides clear licensing terms.

WHICH JDKS DO YOU USE INSIDE YOUR CONTAINERS?



69% OF RESPONDENTS REPORT USING JDKS THAT ARE SHIPPED WITH KNOWN CVES

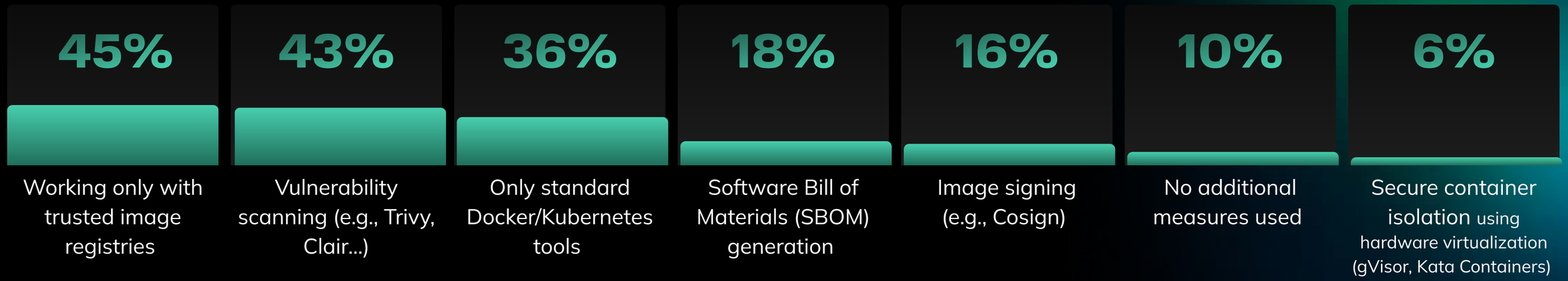
ORGANIZATIONS REMAIN STUCK IN REACTIVE SECURITY POSTURE

Responses reveal different security maturity stages. Trusted registries at 45% and vulnerability scanning at 43% represent basic hygiene—catching known vulnerabilities reactively. SBOM generation at 18% and image signing at 16% address supply chain visibility but remain challenging to implement. Only 6% use hardware isolation.

Ten percent report no additional security measures beyond standard tools.

The focus remains heavily weighted toward detection and monitoring rather than prevention. Most organizations are trapped in reactive security, constantly responding to newly discovered vulnerabilities rather than building on foundations that minimize exposure from day one.

WHAT CONTAINER SECURITY MECHANISMS DO YOU CURRENTLY APPLY?



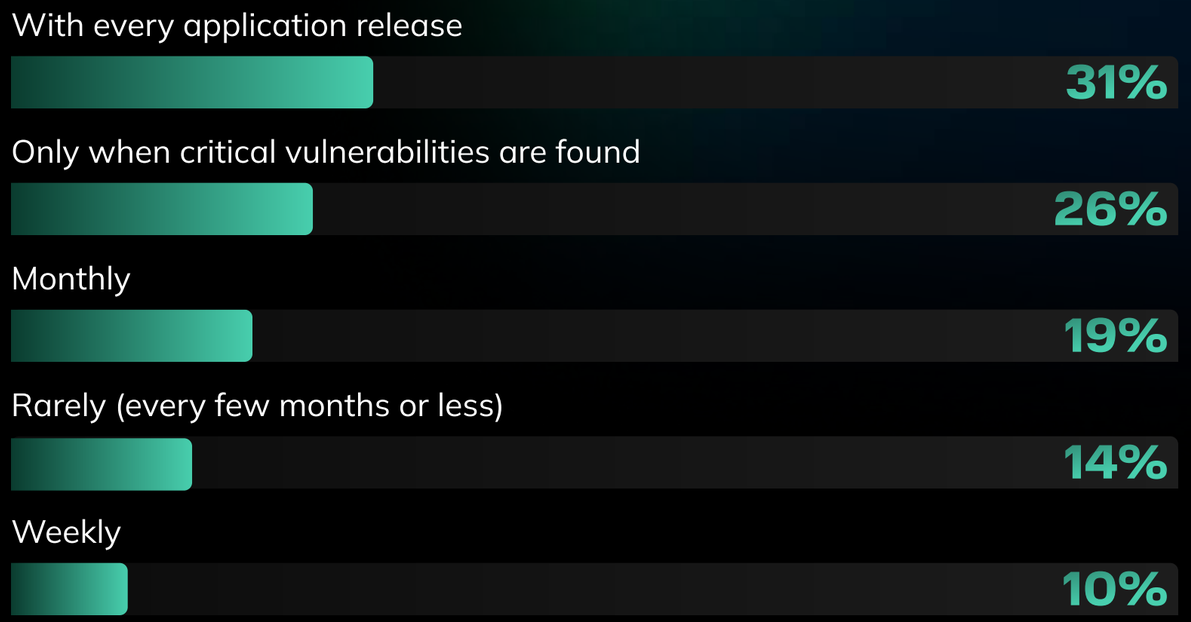
DANGEROUS UPDATE DELAYS LEAVE KNOWN VULNERABILITIES IN PRODUCTION

While 31% update with every release and 26% when critical vulnerabilities emerge, 33% update monthly, rarely, or only a few times yearly. In today's threat landscape, this represents substantial risk.

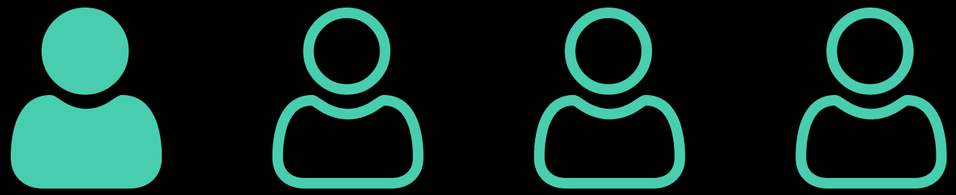
Time between vulnerability disclosure and exploitation continues shrinking. What once took weeks now happens in days. Organizations updating monthly or less operate with known vulnerabilities attackers actively exploit.

The reasons are understandable: updates require testing, validation, and coordination. But this creates a dangerous pattern. The longer you delay, the more changes accumulate, making each update riskier—a negative feedback loop that further discourages frequent updates.

HOW OFTEN DO YOU UPDATE YOUR CONTAINER IMAGES?



ONE IN FOUR HIT BY SECURITY INCIDENTS



Twenty-three percent reported security incidents—18% minor and 5% major. The problem isn't detection—it's the gap between vulnerability disclosure and remediation. During this window, often weeks or months, organizations operate with known exposures.

The 60% reporting no incidents shouldn't breed complacency. Many haven't experienced attempted attacks yet or lack visibility to detect successful intrusions.

The data reinforces a critical point: reactive security measures aren't sufficient. Organizations need approaches that minimize risks of human mistakes and vulnerability exposure from the foundation.

IN THE PAST 12 MONTHS, HAS YOUR ORGANIZATION EXPERIENCED ANY SECURITY INCIDENTS RELATED TO CONTAINERIZED APPLICATIONS?

- **5% Yes,** major incident(s)
- **18% Yes,** minor incident(s)
- **60% No** incidents

TEAMS DROWNING IN VULNERABILITY MANAGEMENT WORK

The challenges cluster into three related categories.

Human errors dominate at 62%—even with the best tools, implementation failures remain the primary vulnerability. The second cluster revolves around **CVE management:** patching difficulties at 36%, gaps before patches at 32%, and false positives at 29%. The third centers on **resources:** time constraints at 49% and lack of organizational priority at 36%.

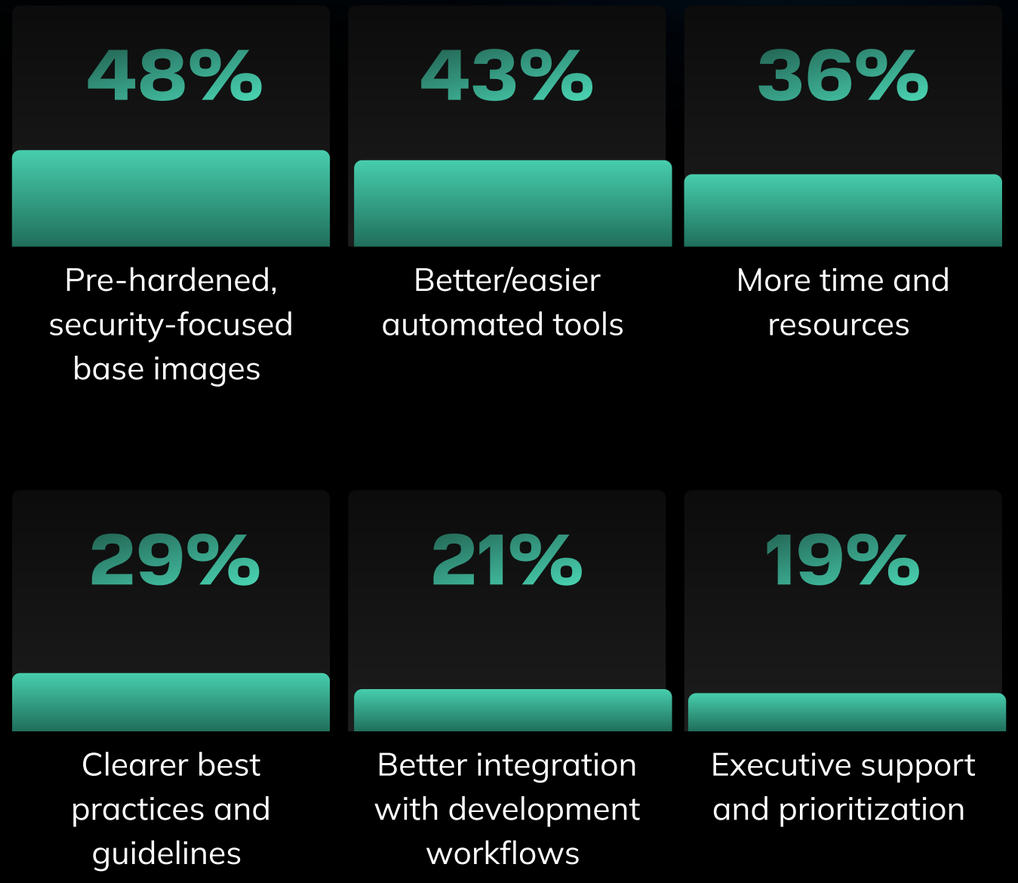
These clusters reinforce each other. Insufficient resources lead to rushed implementations, causing human errors. The overwhelming CVE workload consumes available time, leaving no capacity for proactive improvements.

ORGANIZATIONS NEED RELIEF FROM ENDLESS VULNERABILITY MANAGEMENT

The top three converge on the same need: relief from overwhelming vulnerability management. Pre-hardened images at 48%, better automation at 47%, and more resources at 36% are different approaches to the same problem.

Pre-hardened images shift from reactive to proactive security. The vendor handles all the firefighting—scanning, patching, and testing. Whenever you pull an image from the registry, it's free of known CVEs and stays that way. Patching becomes the vendor's responsibility, completely removing this workload from your teams.

WHAT WOULD MOST HELP YOUR ORGANIZATION IMPROVE CONTAINER SECURITY?



bell//soft

CONCLUSION

Across every section of the survey, one message repeats consistently: **Teams want security, efficiency, and simplicity, but their current tooling makes this difficult to achieve.**

The 48% requesting pre-hardened images aren't just looking for a security tool. They're signaling a desire to build their platform engineering on a foundation that's secure by default. Hardened vendor-maintained images directly address the root causes of today's container security challenges, reducing vulnerability exposure, operational strain, cloud costs, and risk of human errors.

As the industry moves beyond reactive security, hardened images are emerging as the most effective path toward stable, low-maintenance, high-security container environments.