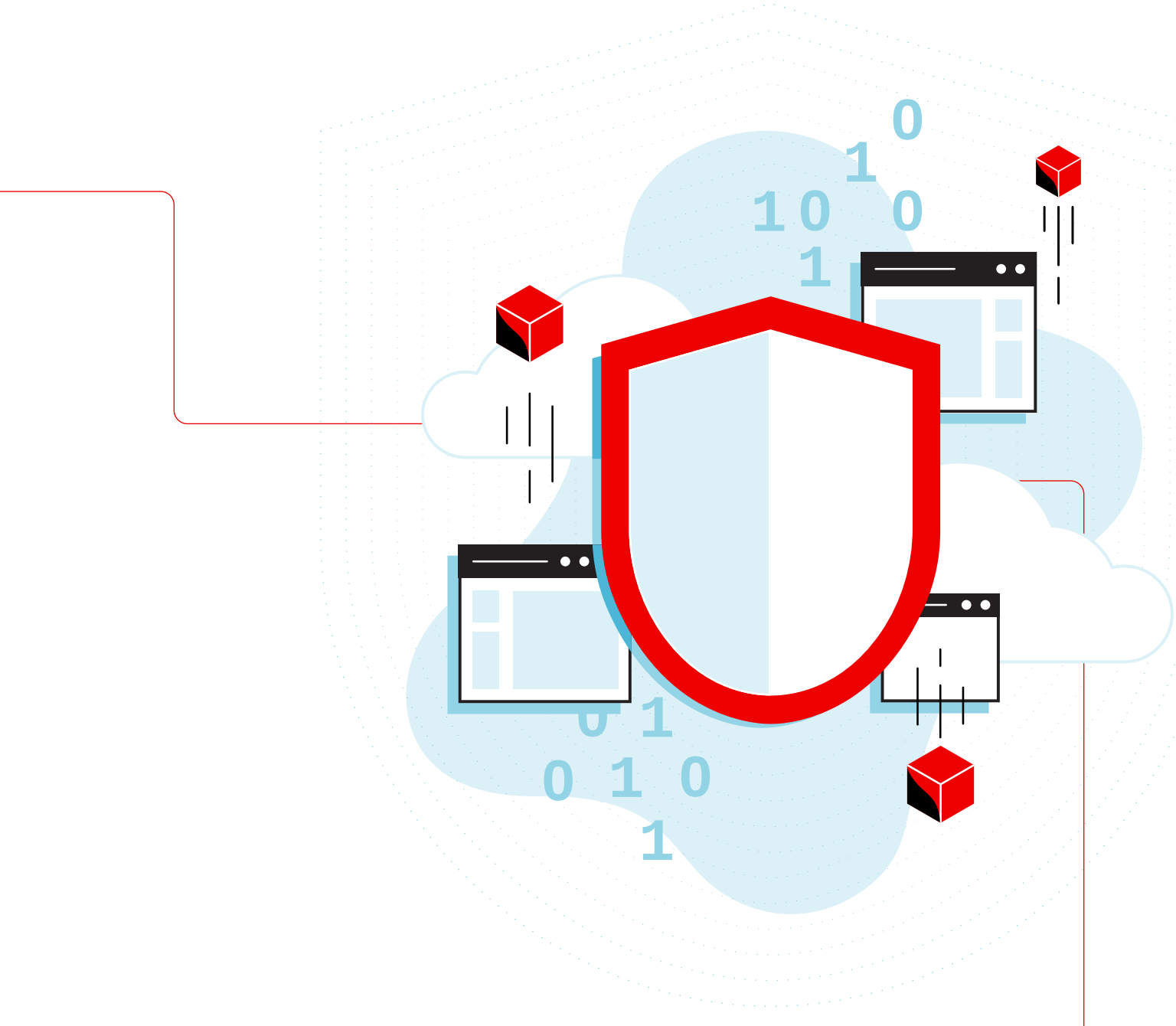
Red Hat

2022

State of Kubernetes security report



Executive summary

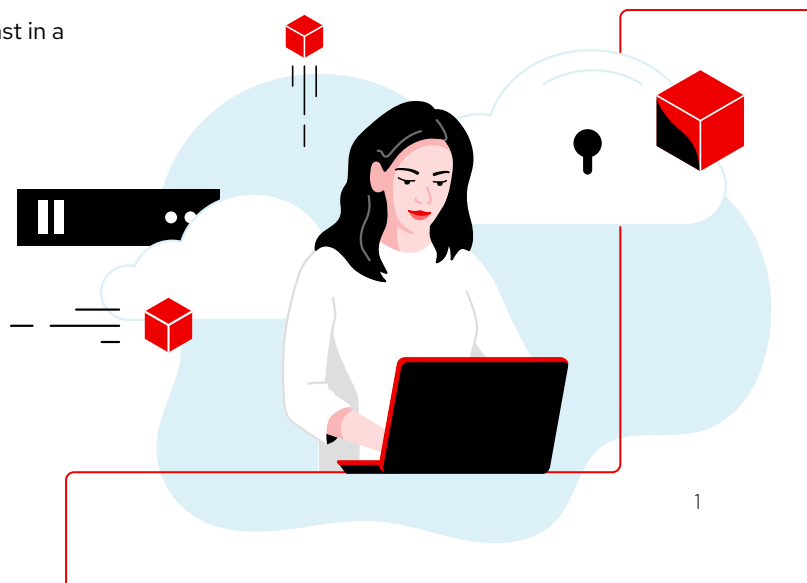
Our latest edition of the State of Kubernetes security report analyzes emerging trends in container, Kubernetes, and cloud-native security.

The report highlights the security challenges in cloud-native development—and how organizations are addressing the challenges and protecting their applications. Based on survey results from more than 300 DevOps, engineering, and security professionals, this report uncovers new findings about how companies that embrace containers and Kubernetes implement DevSecOps initiatives to protect their cloud-native environments.

Security is one of the biggest concerns with container adoption, and security issues continue to cause delays in deploying applications into production. We also look at the most common types of security incidents that companies experience in their Kubernetes environments, as well as whether they report any customer or revenue loss due to these incidents.

DevSecOps is quickly becoming a standard for shifting security left and addressing security issues within the DevOps workflows, with over 3/4 of respondents having initiatives that increase collaboration between DevOps and Security teams. The survey results highlight the importance of collaboration across Dev, Ops, and Security teams to implement security early in the development life cycle to realize the greatest benefit of Kubernetes—innovating fast.

We encourage you to benchmark yourself against the findings in this report to determine how you can accelerate your efforts to apply security controls across containers and Kubernetes. Delaying security could mean delaying innovation or facing financial loss. There are many security advantages you can use in containers and Kubernetes—from declarative configuration and immutable infrastructure to the isolation inherent in containerized applications. Organizations, however, need the knowledge, tooling, and processes to put those capabilities to work so they can benefit from the sizable advantages of running fast in a DevOps-driven, cloud-native world.



Key findings

53%
Detected a misconfiguration in Kubernetes in last 12 months

57%
Worry the most about securing workloads at runtime

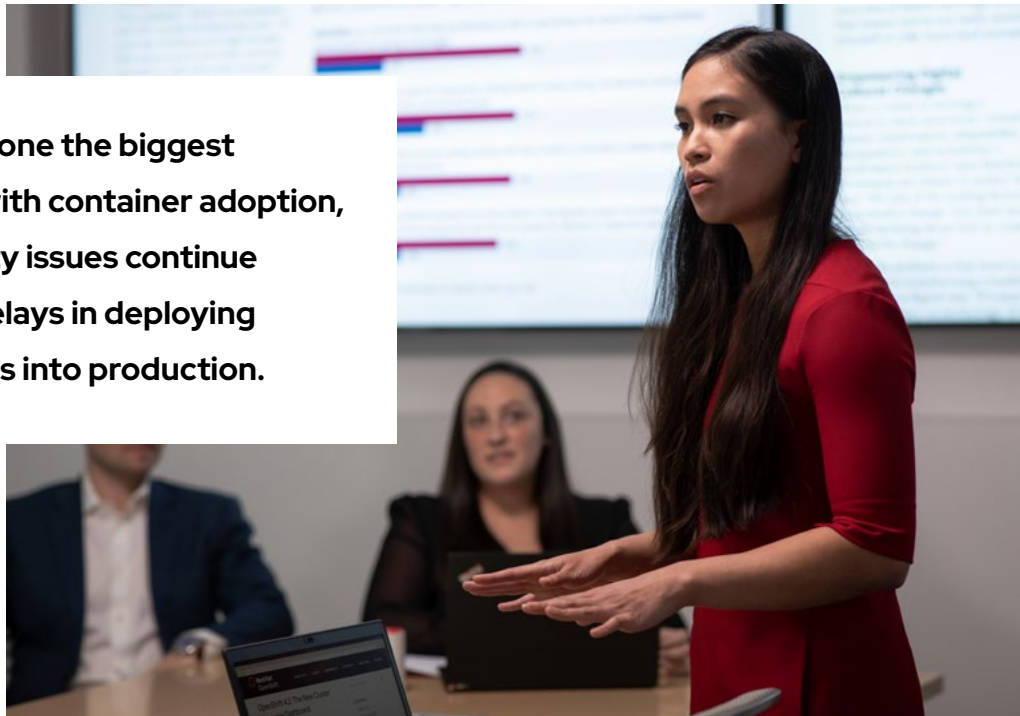
51%
Require developers to use validated images

78%
Have a DevSecOps initiative in either beginning or advanced stages

43%
Consider "DevOps" as the role most responsible for Kubernetes security

55%
Delayed or slowed down application deployment due to security concern

Security is one the biggest concerns with container adoption, and security issues continue to cause delays in deploying applications into production.



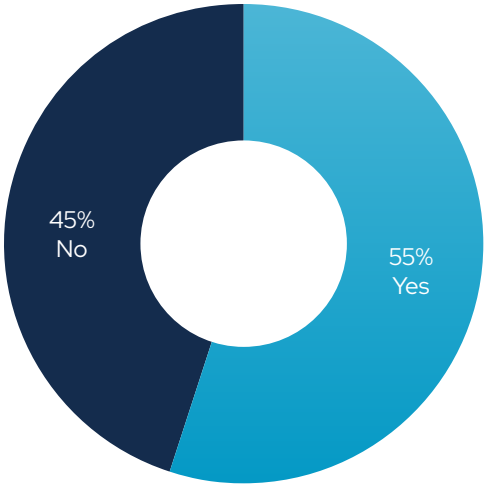
Security concerns hinder innovation

Majority of respondents experience slowdown in application delivery due to unaddressed security concerns.

Organizations are transforming how they build, run, and scale applications by adopting cloud-native technologies like Kubernetes and microservices-based application architectures. Some are building all new applications as microservices while others refactor existing applications alongside managing monoliths. They must also transform how they implement security across their cloud-native stack.

More rapid release cycles, faster bug fixes, and greater flexibility to run and manage applications across hybrid environments are three of the most often cited benefits of containerization. However, when security becomes an afterthought, you may end up negating the greatest gain of containerization—agility. The majority of survey respondents (55%) have had to delay an application rollout because of security concerns over the last 12 months. New technologies often create unforseen security challenges. Some organizations are overwhelmed by security needs that stretch across all aspects of the application life cycle, from development through deployment and maintenance. They need a simplified way to protect their containerized applications without slowing development, or increasing operational complexity.

Have you ever delayed or slowed down application deployment into production due to container or Kubernetes security concerns?

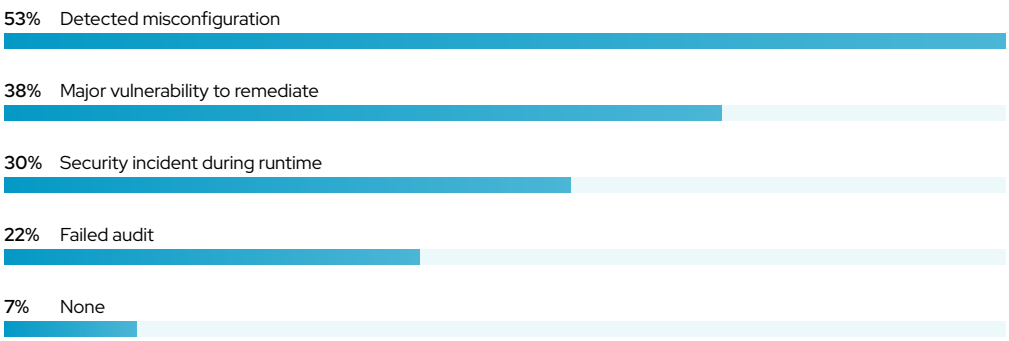


93% of respondents experienced at least one security incident in their Kubernetes environments in the last 12 months, sometimes leading to revenue or customer loss.

93% of respondents have experienced a security incident in their Kubernetes and container environments during the last 12 months. A combination of factors are likely behind this, including a lack of security knowledge about containers and Kubernetes, inadequate or unfit security tooling, and central security teams unable to keep up with fast-moving application development teams. As a consequence, 31% of respondents say they have experienced revenue or customer loss due to a security incident over the last 12 months.

Human error continues to lead the causes of data breaches. A recent study revealed that human error was a major contributing factor in 95% of breaches.[1] Kubernetes and containers, while powerful, were designed for developer productivity, not necessarily security. Default pod-to-pod network settings, as an example, allow open communication to quickly get a cluster up and running, at the expense of security hardening. Not surprisingly, nearly 53% of respondents have experienced a misconfiguration incident in their environments over the last 12 months. 38% have discovered a major vulnerability, and 30% said they have suffered a runtime security incident. Lastly, 22% said they had failed an audit.

In the past 12 months, what security incidents or issues related to containers and/or Kubernetes have you experienced? (pick as many as apply)



In the last 12 months, have you experienced revenue/customer loss due to a container/Kubernetes security or compliance issue/incident?

1. World Economic Forum. "The Global Risks Report 2022," Jan. 2022.

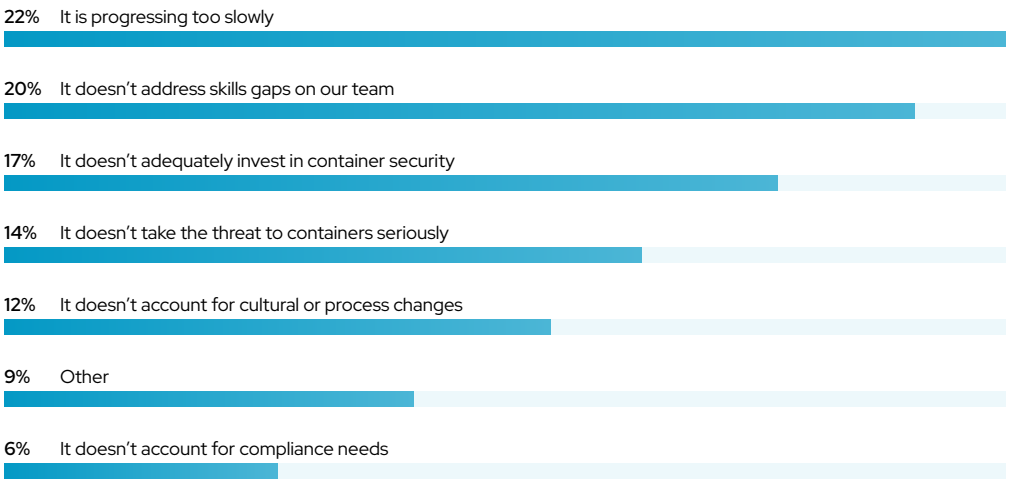
Security is one of the top concerns with container strategies

Since our earlier finding underscored the prevalence of security incidents in these environments (93%), it should come as no surprise that security remains a top concern when it comes to container strategies.

Taken together, concerns about security threats to containers (14%) and a lack of investment in container security (17%) shows that security is the top concern with container strategies, at 31%. Another 22% of respondents cited slow pace of progress as their biggest concern, while 20% consider skills gaps as their biggest concern.

Organizations are eagerly adopting containers and Kubernetes and investing in the cloud-native ecosystem to foster innovation and growth. If they do not make the necessary investments in security strategies and tooling simultaneously, they risk the security of their critical applications and may even need to delay application rollout.

What is your biggest concern about your company's container strategy?



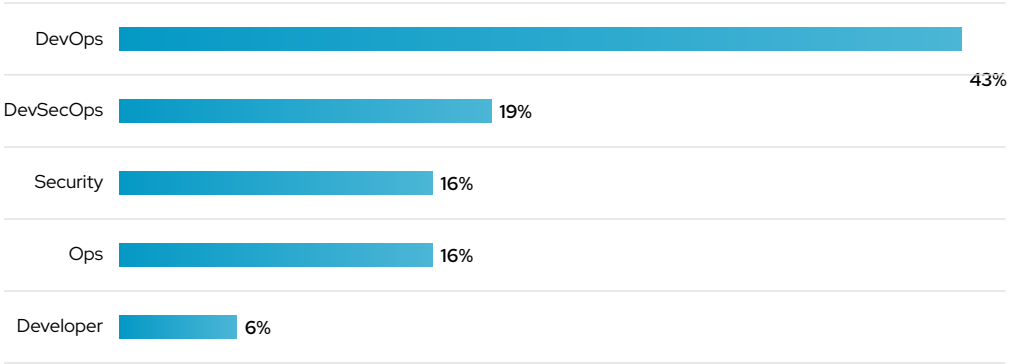
Responsibility for Kubernetes security remains decentralized, with DevOps taking the leading role

Across various roles, DevOps is the single role most cited as responsible for securing containers and Kubernetes.

Taken together, the myriad operational roles of DevOps, Ops, and DevSecOps are considered the primary owners of Kubernetes security by a whopping 78% of respondents.

Only 16% of respondents identify the central IT security team to hold responsibility for Kubernetes security. One way to explain this is by looking at what motivates container and Kubernetes adoption. Adoption is often primarily motivated by DevOps as shadow IT and bypasses central IT team governance, so it is not surprising to see respondents naming them responsible for securing these technologies. To bridge these gaps, container and Kubernetes security tooling must facilitate close collaboration among different teams—from Developers to DevOps to Ops to Security—instead of perpetuating team isolation that may plague organizations.

What role at your organization is most responsible for container and Kubernetes security?

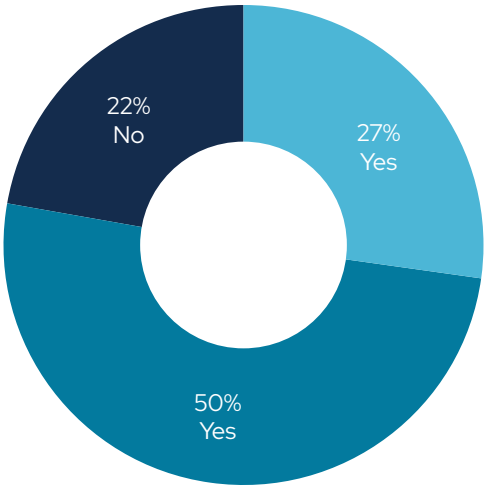


DevSecOps is seeing mass adoption

A majority embraces DevSecOps—a term that encompasses the processes and tooling that allows security to be built into the application development life cycle, rather than as a separate process.

Our survey found encouraging news—the vast majority of respondents say they have some form of DevSecOps initiative underway. Only 22% of respondents continue to operate DevOps separate from Security. 27% of respondents count themselves among the most forward-looking organizations when it comes to DevSecOps, with an advanced DevSecOps initiative, where they are integrating and automating security throughout the life cycle.

Do you have a DevSecOps initiative in your organization?



27% Yes
It's in an advanced stage, where we're integrating and automating security throughout the life cyle

50% Yes
It's in an early stage, with DevOps and Security collaborating on joint policies and workflows

22% No
DevOps and Security remain separate, with minimal collaboration

Respondents worry about misconfigurations above all other security concerns

Kubernetes is a highly customizable container orchestrator, with various configuration options that affect an application's security posture.

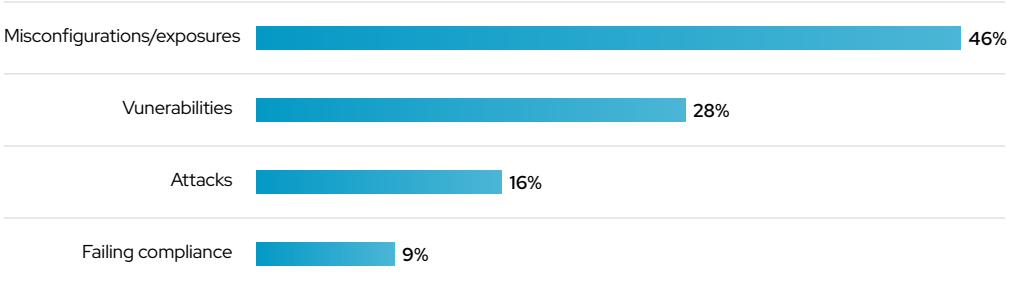
Consequently, respondents worry the most about exposures due to misconfigurations in their container and Kubernetes environments (46%)—nearly three times the level of concern over attacks (16%), with vulnerabilities as the second-leading cause of worry (28%).

Configuration management poses a uniquely difficult challenge for security practitioners. The persons responsible for configuring workloads may not understand security implications of various settings within Kubeneretes. While a host of tools are available for vulnerability scanning of container images, configuration management requires more consideration and will likely be unique to organizations and teams depending on their risk tolerance and level of workload sensitivity.

People may know that they should avoid deploying the Kubernetes dashboard, but configuring a pod's security context or implementing Kubernetes role-based access control (RBAC) are just two examples of more challenging settings that teams need to get right.

The best way to address this challenge is to automate configuration management as much as possible, so that security tools—rather than humans—provide the guardrails that help developers and DevOps teams configure containers and Kubernetes more securely.

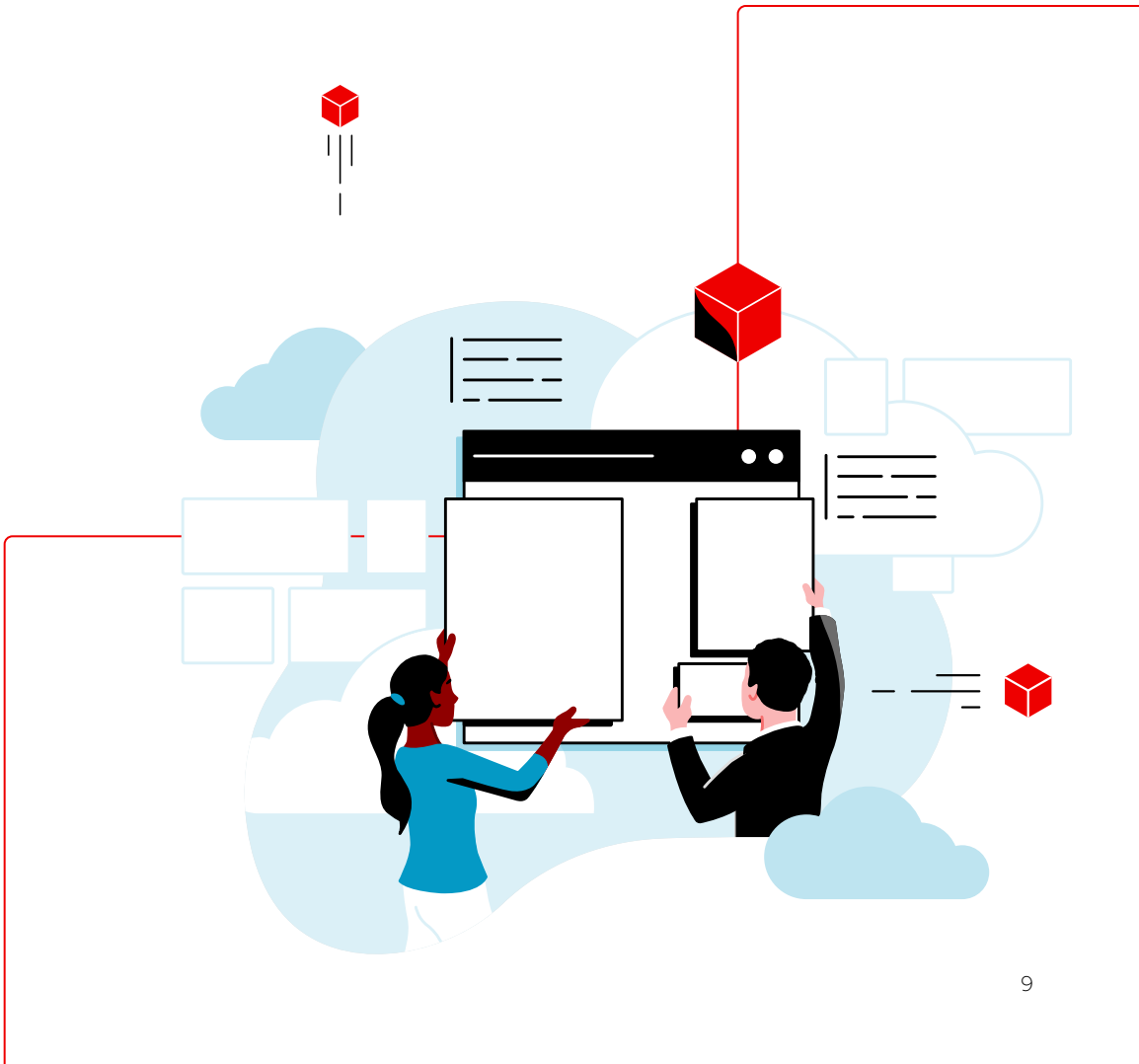
Of the following risks, which one are you most worried about for your container and Kubernetes environments?



57% of respondents worry the most about the runtime phase of the container life cycle

Runtime—sometimes also known as Day 2 operations or the post-deployment stage—is the container life cycle phase that organizations worry about the most. This concern seems counterintuitive given that an overwhelming majority of respondents identify misconfigurations as the source of biggest security risk and have experienced a misconfiguration incident more often than any other type of security incident.

However, consider that runtime security issues are usually caused by security lapses—such as a misconfiguration—at build or deploy stage. Furthermore, any negative outcome of a security misstep at build or deploy stages is likely to be felt only once an application is running in production. Incident response, a key aspect of security, is also more complicated at runtime. Lastly, security issues discovered at runtime are likely more costly to fix as well. All together, it makes heightened runtime security worries more understandable.



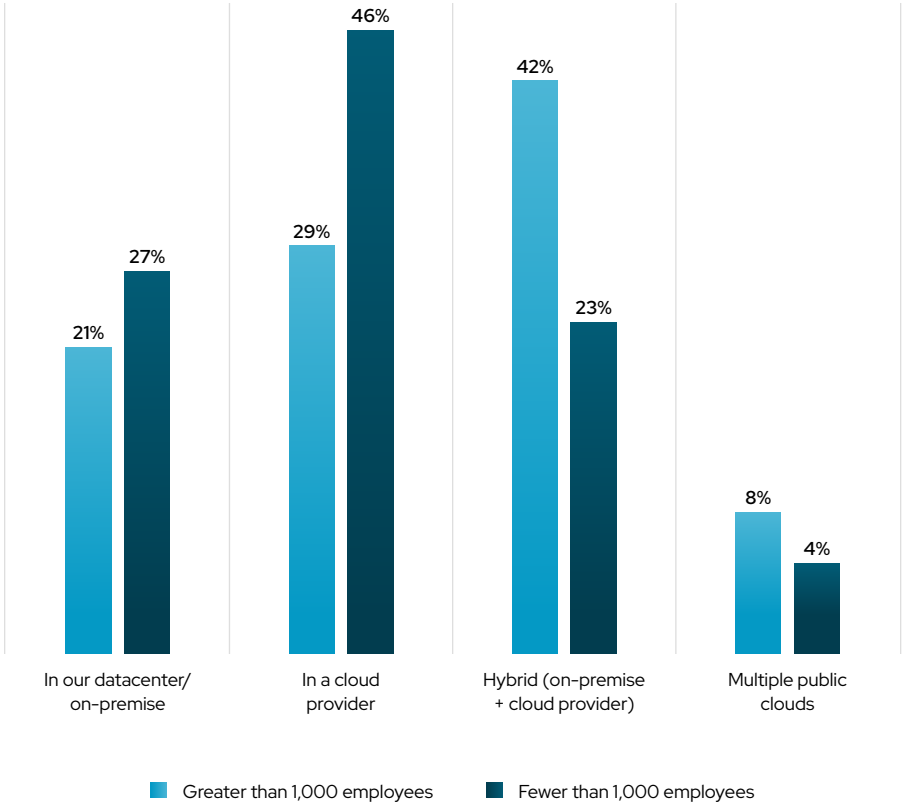
Hybrid cloud deployment strategies preferred by large organizations

Talk of cloud-only strategies runs high, but actual deployments on one or multiple cloud providers is highly correlated to the size of the organization.

Larger organizations (1,000+ employees) favor a hybrid approach for running containerized applications (42%), while smaller organizations gravitate toward a single cloud strategy (46%).

With hybrid models continuing to be the dominant approach for enterprises and other large organizations, they need consistent security and compliance.

Where do you have containers running?



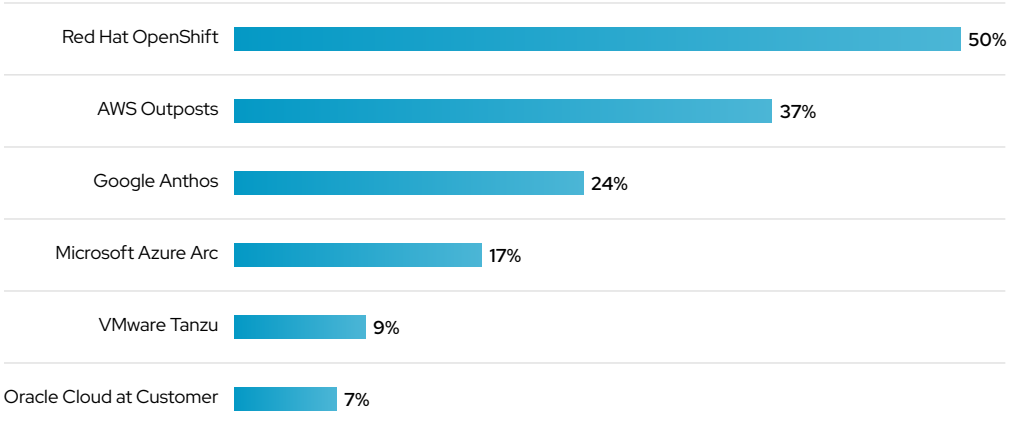
Red Hat OpenShift is the leader in hybrid cloud deployments

With hybrid cloud deployments being the most popular mode of running containerized applications at large organizations, we wanted to understand how organizations were deploying in hybrid mode.

Half of all respondents running containerized workloads in hybrid environments use Red Hat® OpenShift® to manage their containers and Kubernetes.

The popularity of technologies from public cloud providers follows a similar arc of overall platform popularity, with 37% of respondents using AWS Outposts, 24% using Google Anthos, and 17% using Azure Arc. The hybrid offerings from VMware and Oracle lag behind their peers, with 9% and 7% of respondents using them, respectively.

Are you using any solutions for hybrid and multicloud Kubernetes deployments? (pick as many as apply)



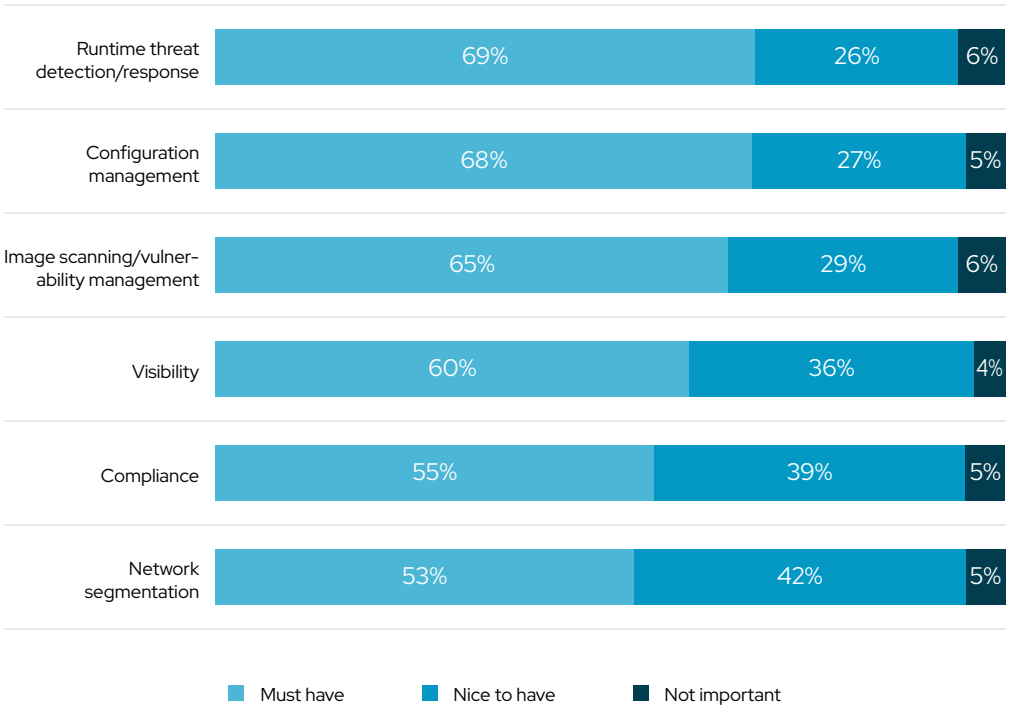
Top Kubernetes security use cases

Kubernetes security use cases span the entire application development life cycle, and organizations expect a security solution that protects containers and Kubernetes at every phase.

The capabilities span DevOps and security activities, underscoring the need for both broad and deep functionality in container and Kubernetes security platforms. This breadth also highlights the fact that securing Kubernetes and containers requires involvement from Dev, Ops, and Security teams.

Respondents identified runtime threat detection/response, configuration management, and image scanning/vulnerability management as the three most important Kubernetes security use cases to address, with 69%, 68%, and 65% of respondents identifying each as a "must-have" capability, respectively.

How would you rate the importance of the following Kubernetes security capabilities?



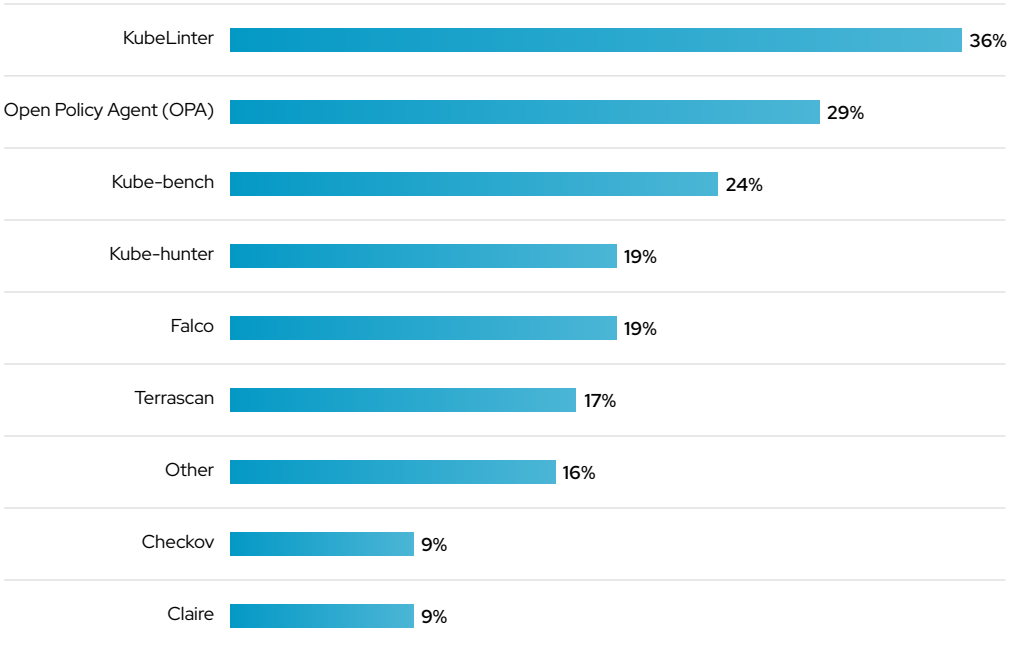
Open Policy Agent (OPA) and KubeLinter are two of the most-used open source solutions for Kubernetes security

Alongside commercial Kubernetes security products, our respondents rely on a number of open source security tools to protect their cloud-native applications.

KubeLinter, an open source YAML and HELM linter for Kubernetes, is used by 36% of respondents, while 29% say they use OPA for Kubernetes security.

Customers have a rich selection of open source security tools to choose from, and our survey results show that no single open source security tool dominates the Kubernetes security market. Nearly a quarter of respondents also use Kube-bench, Kube-hunter, and Falco each.

What open source tools do you use for Kubernetes security? (pick as many as apply)



4 tips for achieving better security

When security is ignored, organizations are putting at risk the core benefit of faster application development and release by not ensuring that their cloud-native environments are built, deployed, and managed securely.

Our findings show that what happens in the build and deploy stages has a significant impact on security, which was underscored by the prevalence of misconfigurations and vulnerabilities across organizations. Security, therefore, must shift left, imperceptibly embedding into DevOps workflows instead of being "bolted on" when the application is about to be deployed into production.

1 Use Kubernetes-native security architectures and controls.

Kubernetes-native security uses the rich declarative data and native controls in Kubernetes to deliver several key security benefits. Analyzing the declarative data available in Kubernetes yields better security, with risk-based insights into configuration management, compliance, segmentation, and Kubernetes-specific vulnerabilities. Using the same infrastructure and its controls for application development and security reduces the learning curve and supports faster analysis and troubleshooting. It also eliminates operational conflict by ensuring security gains the same automation and scalability advantages that Kubernetes extends to infrastructure.

2 Security should start early but extend across the full life cycle, from build/deploy to runtime.

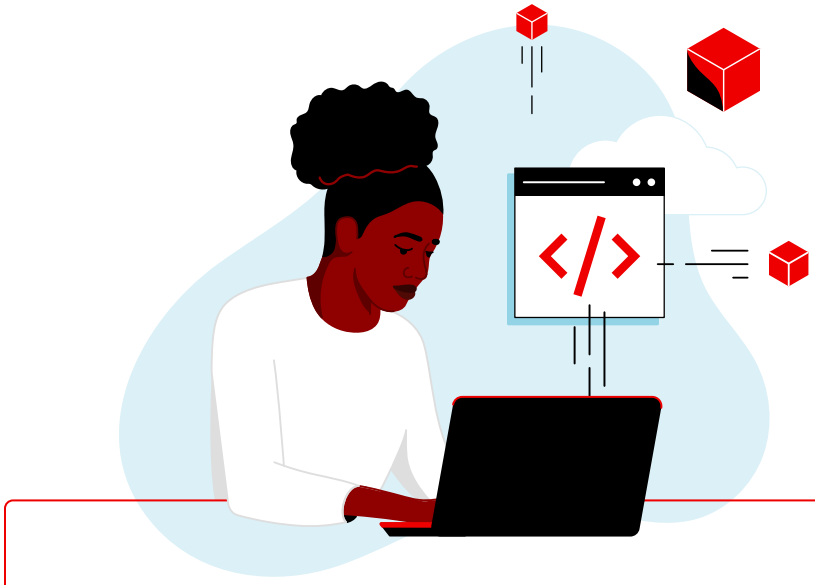
Security has long been viewed as a business inhibitor, especially by developers and DevOps teams whose core mandates are to deliver code fast. With containers and Kubernetes, security should become a business accelerator by helping developers build strong security into their assets right from the start. Look for a container and Kubernetes security platform that incorporates DevOps best practices and internal controls as part of its configuration checks. It should also assess the configuration of Kubernetes itself for its security posture, so developers can focus on feature delivery.

3 Require portability across hybrid environments.

With most organizations deploying containers in both on-premise and public cloud environments (sometimes in multiple clouds), security must apply consistently wherever your assets are running. The common foundation is Kubernetes, so make Kubernetes your source of truth, your point of enforcement, and your universal visibility layer so you have consistent security. Managed Kubernetes services may quicken your organization's ability to adopt Kubernetes, but be careful about getting locked into cloud provider-specific tooling and services.

4 Transform the developer into a security user by building a bridge between DevOps and Security.

Given most organizations expect DevOps to run container security platforms, your security tooling must help bridge Security and DevOps. To be effective, the platform must have security controls that make sense in a containerized, Kubernetes-based environment. It should also assess risk appropriately. Telling a developer to fix all 39 discovered vulnerabilities with a Common Vulnerability Scoring System (CVSS) score of seven or higher is inefficient. Identifying for that developer the three deployments that are exposed to that vulnerability, and showing why they are risky, will significantly improve your security posture.

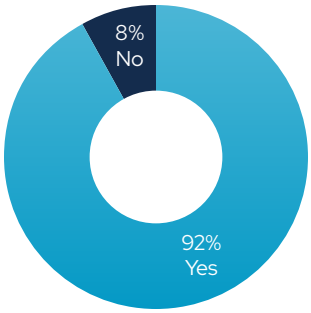


About our respondents

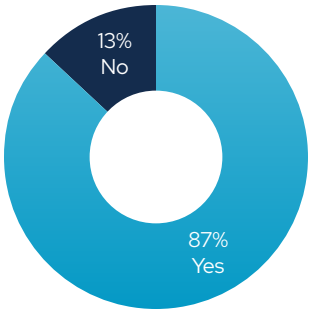
Kubernetes service

92% of respondents use Kubernetes (87% as production workloads), with Amazon EKS, Red Hat OpenShift, and self-managed Kubernetes as the three most popular Kubernetes services.

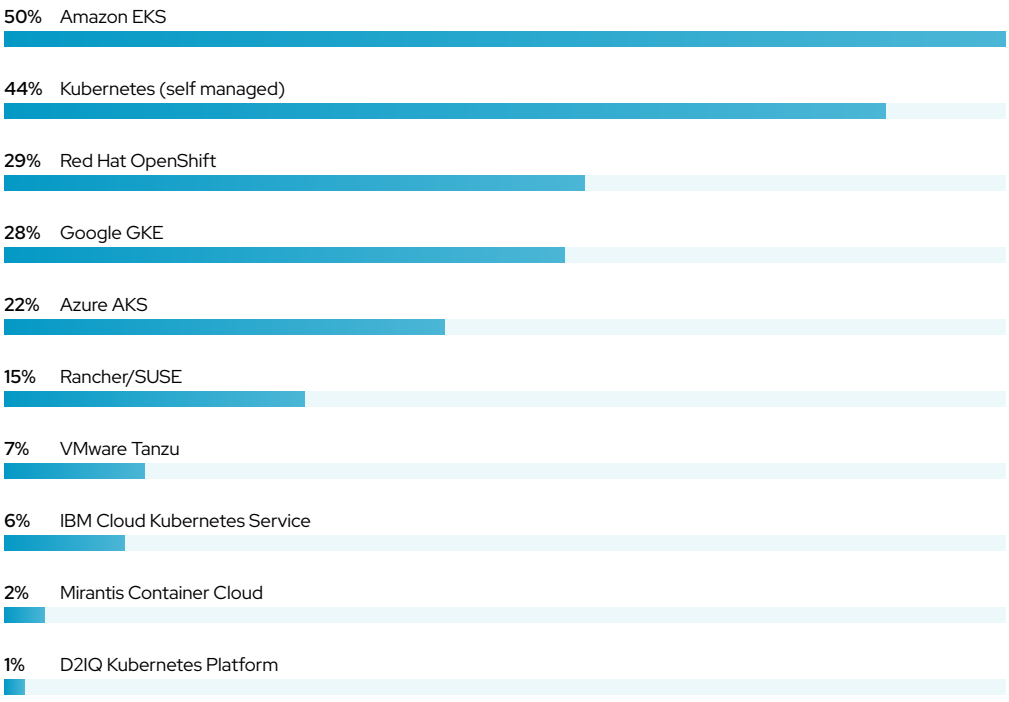
Are you using Kubernetes for container Orchestration?



Are you running any production workloads on Kubernetes?



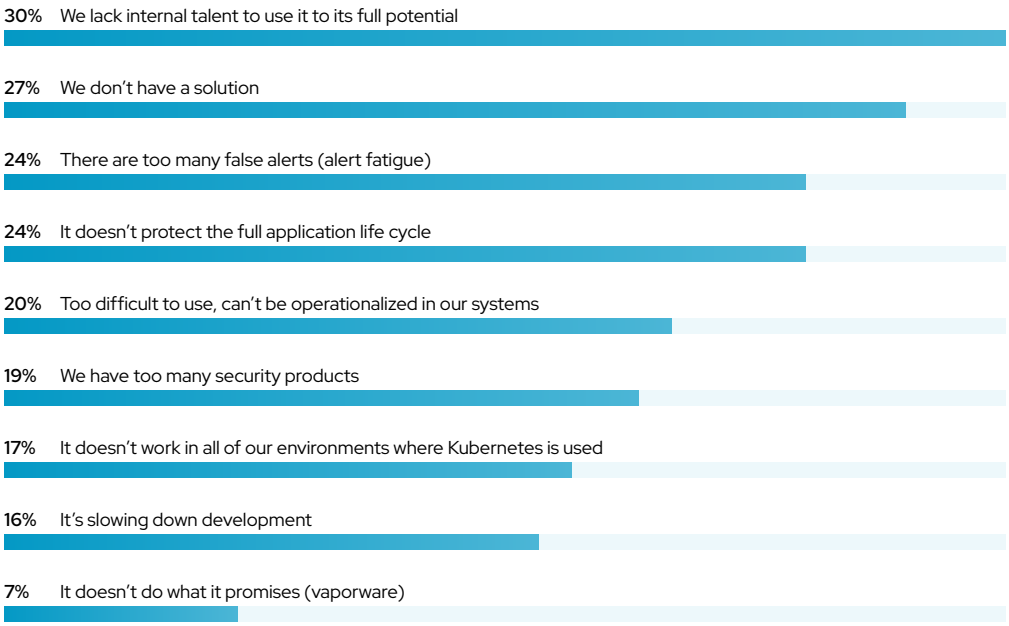
What Kubernetes platform do you use to orchestrate your containers? (pick as many as apply)



Common pain points with Kubernetes security vendors

Lack of internal talent and too many false positives are two of the most common pain points with Kubernetes security solutions in the market.

What is your biggest pain point with your current Kubernetes security solution? (Pick as many as apply)

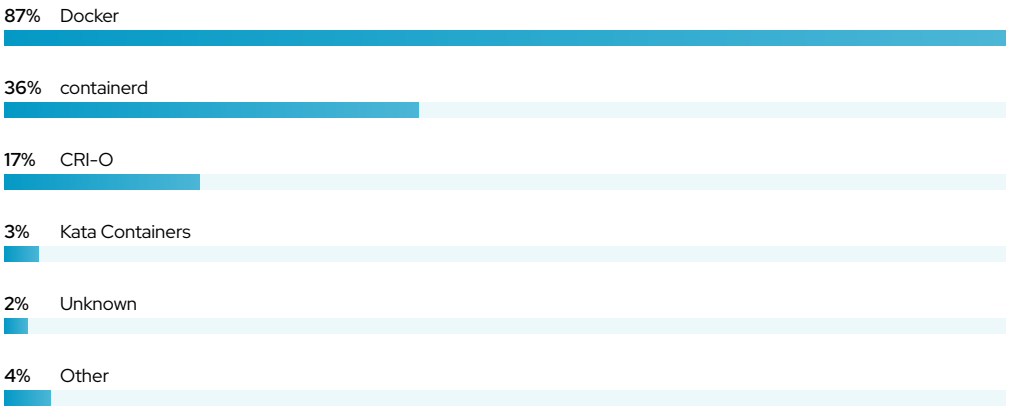


Edge deployments

Nearly half of respondents run containers at edge locations.

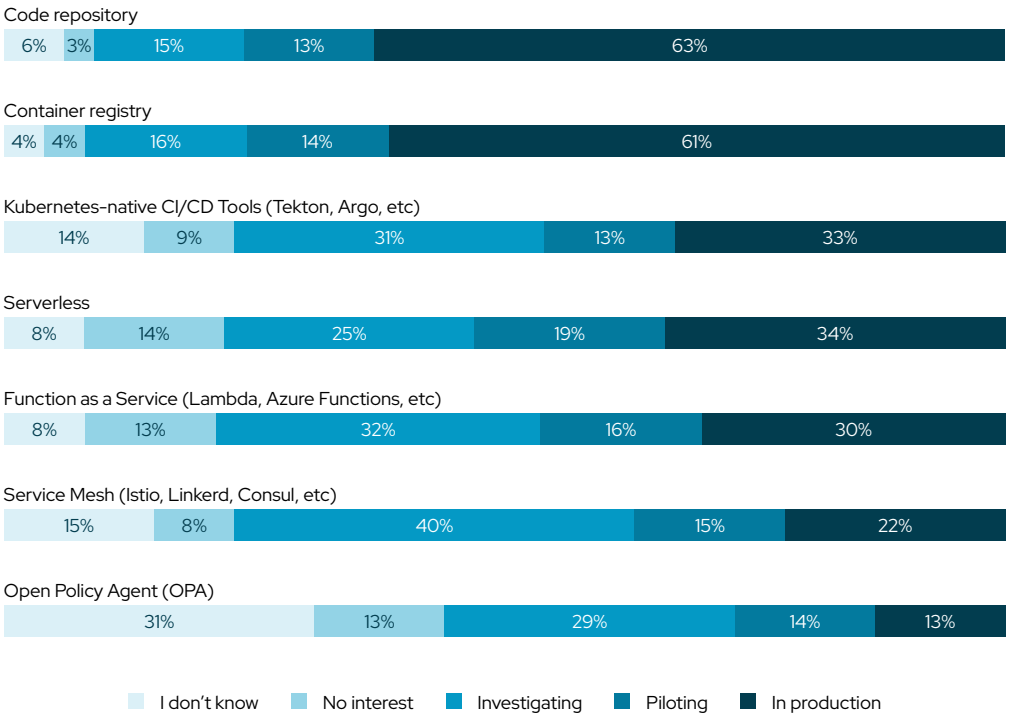
Container runtime technology

Docker runtime engine remains dominant, with containerd a distant second but closing the gap.

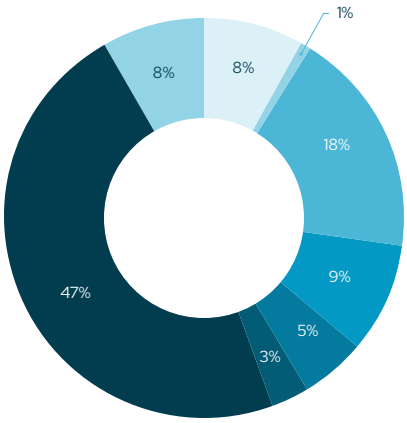


Other cloud-native technologies

Not counting code repositories and container registries, emerging cloud-native technologies are still in early adoption stages.

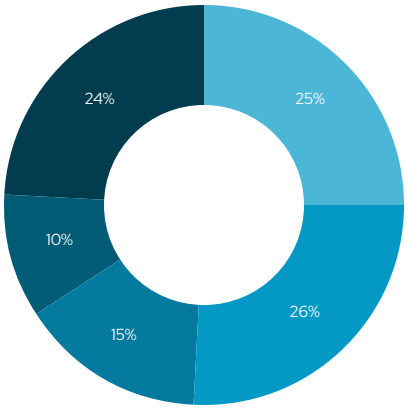


Core demographics



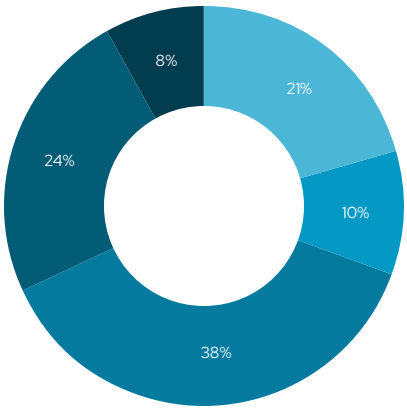
Industry

8% Education
1% Entertainment
18% Financial services/insurance
9% Healthcare
5% Manufacturing
3% Media
47% Technology
8% Other



Company size

25% 1–100
26% 101–1,000
15% 1,001–5,000
10% 5,001–10,000
24% 10,001+



Functional job role

21% Security
10% Compliance/risk
38% Operations
24% Product development/engineering
8% Other

Learn more about Red Hat Advanced Cluster Security for Kubernetes

Red Hat Advanced Cluster Security for Kubernetes is a Kubernetes-native container security platform that protects your application across build, deploy, and runtime as you progress on your container journey.

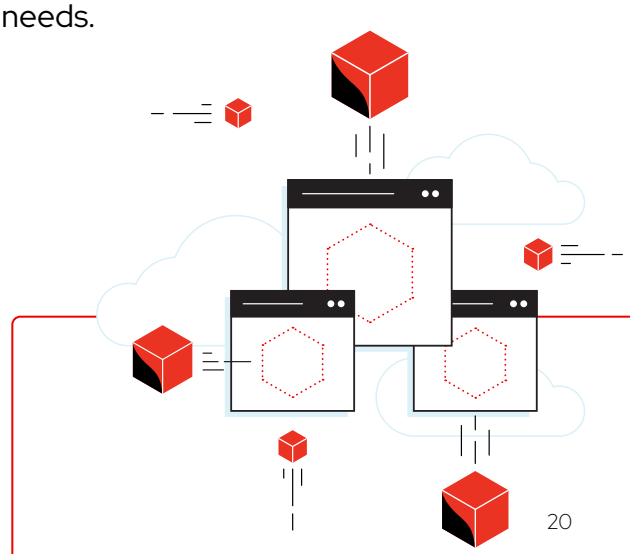
As your environment grows more complex and you depend on more automation, our platform will let you operationalize security in those more sophisticated environments and keep pace with the speed of DevOps.

Kubernetes-native security provides the following crucial benefits.

- **Minimize operational risk:** Align security with DevOps by using Kubernetes-native controls to mitigate threats and enforce security policies that minimize operational risk to your applications.
- **Reduce operational cost:** Reduce the overall investment in time, effort, and personnel, and streamline security analysis, investigation, and remediation by using a common source of truth.
- **Accelerate DevOps productivity:** Accelerate the pace of innovation by providing developers actionable and context-rich guardrails embedded into existing workflows and tooling that supports developer velocity.

Ready to see Red Hat Advanced Cluster Security for Kubernetes in action? Get a personalized demo tailored for your business and needs.

Request demo



Red Hat

Thank you for downloading this Red Hat resource! Carahsoft is the Master GSA and SLSA Dealer and Distributor for Red Hat Enterprise Open Source solutions available via GSA, SLSA, ITES-SW2, The Quilt and other contract vehicles.

To learn how to take the next step toward acquiring Red Hat's solutions, please check out the following resources and information:

For additional resources:
carah.io/RedHatResources

For upcoming events:
carah.io/RedHatEvents

For additional Red Hat solutions:
carah.io/RedHatPortfolio

For additional Open Source solutions:
carah.io/OpenSourceSolutions

To set up a meeting:
redhat@carahsoft.com
877-RHAT-GOV

To purchase, check out the contract vehicles available for procurement:
carah.io/RedHatContracts