WIZ

5 Signs You Need a New Container Security Solution

If you deploy cloud-native apps, you probably use Kubernetes. In fact, as of 2021, 96 percent of organizations were either using or evaluating Kubernetes, according to a Cloud Native Computing Foundation (CNCF) survey, and no fewer than 5.6 backend engineers – or about 31 percent of all developers – were deploying apps on K8s.

It's a safe bet, though, that not all of those organizations are properly managing Kubernetes security or following all container hardening best practices. Although they may have some container security solutions in place, the pace with which many teams have adopted containers and Kubernetes means that traditional security solutions don't always deliver the protections necessary to harden containerized apps in an efficient, scalable way.

To prove the point, let's walk through five common signs that your container security solution is falling short, and that it's time to look for a new approach to securing your containerized apps and Kubernetes clusters.

Is it time for a new approach?

The indicators that you should rethink your container security strategy or implementation of Kubernetes security best practices boil down to:

- Lack of visibility and context on container security risks.
- A siloed approach to container security.
- Trouble managing compliance needs.
- Insufficient threat detection and response abilities.
- Lack of democratization – meaning you struggle to ensure that all stakeholders (developers, security teams and beyond) can access and benefit from container security insights.

Let's take a deeper look at the 5 signs you should look out for:

1. Limited container security visibility and context

In organizations that practice continuous delivery – which about 80 percent of cloud-centric companies do today – Kubernetes-based containerized apps may be deployed as often as once every hour or so by developers and DevOps engineers.

That means that the ability to collect as much data as possible about each application deployment and the environment it's running in is critical for detecting and interpreting security risks. You can't protect your containers if you lack visibility into all layers of your hosting stack – from the underlying infrastructure to the Kubernetes control plane and plugins, to the OS running on each node, to the broader cloud environment of which all of these things are a part.

Unfortunately, not all Kubernetes security tools provide that visibility. Some look only at the local environment in which a container runs, depriving teams of the ability to understand what is happening at other layers of the stack. Others might focus on the Kubernetes environment but lack application- and node-specific visibility.

If this is the case, you need a security solution that provides more complete visibility and context about all levels of your application hosting environment. Otherwise, you run the risk of failing to gain holistic understanding of how a security issue in one part of your stack (like a compromised container running on an EKS cluster) could impact other parts of your environment (like an AWS RDS database).

2. Siloed Kubernetes security tooling and processes

Lack of comprehensive visibility sometimes leads businesses to deploy multiple security tools each designed to handle a different layer of the stack. For example, you might have separate cloud-centric, cluster-centric and configuration-centric security solutions.

While this approach might help you close the visibility gap, it comes at the expense of a consolidated, well-integrated Kubernetes security solution. You end up with multiple information silos, making it hard to correlate security data from different layers of your stack. You're also likely to find teams inundated with alerts, many of which result from the same root cause but are hard to correlate without significant manual effort.

This approach is, in short, a mess. It leads to high-effort, low-impact security processes. If this describes your Kubernetes security solution, you need a new approach that offers better integration and centralized visibility.

3. Container security compliance struggles

Many of the container security tools available today are designed to help automate compliance by assessing how well a deployment conforms to the Center for Internet Security's Docker Benchmarks or similar compliance standards.

Unfortunately, not all solutions do this in an efficient way. Too often, they rely on agents that are deployed on each Kubernetes node to discover misconfigurations and security risks. That leads to high overhead and ties up resources that would be better spent running your actual workloads. You may also struggle to gain timely compliance insights or avoid blindspots due to the time it takes for the agents to detect compliance issues.

If that's the case, you need a Kubernetes security and compliance solution that works in a more scalable, efficient way.

4. Insufficient threat detection and response capabilities

Agent-based container security solutions focus on real-time threat detection and visibility at the cluster level. But because they lack visibility into the broader cloud environment, they don't have complete context into all potential threats. In addition, agent-based threat detection may lead to excess alerts, which causes teams to waste time triaging false positives and reduces their ability to minimize MTTR.

The solution to these challenges is Kubernetes security tooling that can detect threats at all layers and boundaries of your environment, not just the cluster level.

#5. Non-democratized container security

Container security works best when all stakeholders – developers, DevOps engineers, security teams and anyone else who plays a role in detecting or responding to threats – have access to shared solutions that integrate seamlessly with the tools they already use.

When your tools can do this, they enable a democratized approach to container security in which all stakeholders can act as equals. When everyone has access to the same security insights in a way that doesn't slow down their day-to-day workflows, everyone can be a container security hero.

However, many container security tools weren't built with democratized security in mind. They instead take the form of security tools built first and foremost for security teams. Developers and DevOps engineers often see these solutions as a distraction from their primary responsibilities – and something that can disrupt the CI/CD processes they have in place. This leads to a lack of collective responsibility for Kubernetes security and a lower ability to integrate security into all stages of the software delivery process.

The answer to this challenge is to leverage security tools that all stakeholders can love equally.

A better approach to Kubernetes and container security

In short, cloud-native apps and deployment technologies have changed the way organizations develop and deploy applications, but security tools haven't always caught up to the new practices. Too often, security solutions leave businesses with blind spots due to lack of context, force teams to perform manual work that distracts them from other priorities and makes it difficult to take a proactive, comprehensive approach to securing containerized apps.

If this sounds like the way your teams handle container security, it's time to rethink your tooling. You need a Cloud Native Application Protection Platform (CNAPP) that can assess security risks at all layers – from cloud infrastructure, to Kubernetes policies and permissions, to individual application risks and beyond. In addition, CNAPPs deliver contextualized alerts, automated remediation features and proactive incident response capabilities to help teams mitigate risks as rapidly as possible.

To learn more about how to choose a container security solution capable of keeping pace with today's threats and ways of working, download Wiz's Container Security Buyer's Guide, which details what a next-generation container security platform must deliver for modern teams.