Contrast SECURITY

SOLUTION BRIEF

Keeping Kubernetes Secure with Instrumentation

Executive Overview

The DevOps model plays a critical role in the digital transformation of application development. Use of containers has become a popular aspect of DevOps evolution. Whether an application is built in the cloud, on-premises, or in hybrid environments, containerization has clear advantages in terms of scalability, portability, and continuous development and improvement.[1] According to recent survey data, the number of organizations that have containerized at least half of their applications grew by 22% over the previous six months, and organizations with more than half of their containers running in production jumped by nearly one-third.[2]

With containers becoming an increasingly dominant feature in DevOps environments, many organizations are embracing tools to help manage their use of containers. **Kubernetes** offers an open-source platform for managing containerized workloads and automating deployments. Last year, use of Kubernetes as an orchestrator grew from 57% to 86%—a 50% increase.[3]

Unfortunately, containers fail to solve the problem of application security. In fact, the combination of containerization and Kubernetes can increase certain types of risks with legacy approaches to application security. These challenges are resolved with instrumentation that embeds security within the application for always-on observability.

The Nature of Containers Brings New Security Risks

While containers can make it faster and easier to deploy an application, placing an application within a container does not protect the software or make it any less vulnerable to attack. Individual applications still must be assessed and protected within any container environment—including Kubernetes. As Gartner notes, "Container usage for production deployments in enterprises is still constrained by concerns regarding security, monitoring, data management, and networking."[5]

> *According to Gartner, "By 2023, more than 70% of global organizations will be running more than two containerized applications in production, up from less than 20% in 2019."* [4]

Contrast SECURITY

The shared responsibility approach to cybersecurity can be especially challenging when it comes to Kubernetes because security responsibilities are not precisely defined.[6] While the application security mission remains the same (i.e., hardening the environment during the build stage and detecting and remediating threats during runtime), the nature of containerization requires a new approach to security tooling.[7] Indeed, nearly half (44%) of organizations say they have delayed moving an application into production due to security concerns about containers managed with Kubernetes.[8]

> *94% of organizations experienced a security incident in their container and Kubernetes environments within the last 12 months.*[9]

Traditional Application Security Solutions Cannot Keep Up

Many organizations have tried to use existing application security solutions with Kubernetes—adding siloed tools to cover new risk exposures one at a time. But securing containers and their platforms through traditional application security tools is a non-starter. Existing security tooling is not suitable to address the new risks and considerations associated with containers and Kubernetes.[10] And even if the container itself is somehow protected by these means, the container is no longer secure once an insecure application is placed inside it.

Notwithstanding, if an organization chooses a more contemporary approach to AppSec over legacy tools, problems can still ensue. The use of "sidecars" is a modern approach for securing Kubernetes-managed containers—where monitoring agents are placed in their own container alongside the application container. While this is relatively easy to set up in Kubernetes, sidecars can quickly lead to problems with application stability, performance, and security.[11]

Ineffective security methods during development with Kubernetes managed containers also creates bigger problems downstream. Organizations are beginning to realize that many production runtime security failures are caused by missed security best practices in development. For that reason, more than half (57%) of organizations report that they are

more worried about their build and deploy phases, with misconfigurations and vulnerabilities cited as areas posing the greatest risk.[12] Cost is also a critical factor to fixing bugs after the design phase. It is six times more expensive to fix a bug found during implementation; 15 times more if it is identified in testing; and 100 times more once the code is in production.[13]

To address these problems, organizations need to "shift left" and incorporate effective application security from the very beginning— during the build stage of the development cycle. This requires a rethink of traditional application security where coding must be repeatedly halted and restarted for tools like legacy static application security testing (SAST) and dynamic application security testing (DAST).

Container security should leverage the same strengths that containers bring to DevOps processes—it needs to be both dynamic and flexible.[14]

Instrumentation-Based Application Security for Kubernetes

To protect Kubernetes-based containerized applications, organizations must integrate security into the application itself using instrumentation. By placing specialized instrumentation sensors throughout the code itself, organizations can gain comprehensive insights across all parts of the application and embed security across all phases of the software development life cycle (SDLC).

Instrumentation-based application security delivers continuous, automated, real-time identification of vulnerabilities and verification of their remediation. This approach is effective because it operates within the container, protecting the container as well as the application or service hosted inside.[15]

The instrumentation-based Contrast Application Security Platform enables companies to align application security with their container efforts. It supports:

- Extending application security into containers without acquiring/managing more silo-based security tools
- Heterogeneous support for managing containers (e.g., Puppet, Chef, Ansible)
- Multilanguage support for each container
- Coverage of both custom and open-source code
- Fast and easy deployment

The Contrast Application Security Platform Contains Security Risks

It can be challenging to know when an existing security approach needs to be replaced. The high volume of security incidents associated with Kubernetes-managed containers, combined with increasing usage, points to a pressing need for change.

To ensure continuous deployment of containerized code managed by Kubernetes, organizations need an application security platform that extends continuous security assessment and protection across build, deployment, and runtime environments. Contrast's instrumentation-based platform achieves this by managing security from within the application itself while reducing the workflow burdens on limited staff. It reduces costs by helping DevOps find and fix problems during development. It provides embedded security that scales across all parts of the containerized code—including open-source components. Finally, it helps organizations simplify their infrastructure by eliminating dependency on add-on, siloed approaches to application security.

[1] Ali Golshan, "Survey Reveals Rapid Growth in Kubernetes Usage, Security Still a Concern," DZone, August 30, 2019.
[2] Ajmal Kohgadai, "6 Container Adoption Trends of 2020," StackRox, March 4, 2020.
[3] Ali Golshan, "Survey Reveals Rapid Growth in Kubernetes Usage, Security Still a Concern," DZone, August 30, 2019.
[4] Robert Christiansen, "More enterprises are using containers; here's why," CIO, August 26, 2019.
[5] Ajmal Kohgadai, "Gartner best practices for Kubernetes & container security," StackRox, June 25, 2019.
[6] Mike Vizard, "Threat Stack Report Highlights Common Kubernetes Security Issues," Container Journal, April 27, 2020.
[7] Ali Golshan, "Survey Reveals Rapid Growth in Kubernetes Usage, Security Still a Concern," DZone, August 30, 2019.
[8] Ajmal Kohgadai, "5 Surprising Findings from StackRox's Latest Kubernetes Security Report," StackRox, February 19, 2020.
[9] Ajmal Kohgadai, "5 Surprising Findings from StackRox's Latest Kubernetes Security Report," StackRox, February 19, 2020.
[10] Ali Golshan, "Survey Reveals Rapid Growth in Kubernetes Usage, Security Still a Concern," DZone, August 30, 2019.
[11] Apurva Dave, "5 Things We've Learned About Monitoring Containers," DZone, August 14, 2017.
[12] Ali Golshan, "Survey Reveals Rapid Growth in Kubernetes Usage, Security Still a Concern," DZone, August 30, 2019.
[13] Mukesh Soni, "Defect Prevention: Reducing Costs and Enhancing Quality," iSixSigma, accessed April 16, 2020.
[14] Tim Ferrill, "9 container security tools, and why you need them," CSO, August 4, 2020.
[15] Erik Costlow, "Security Concerns Remain with Containers and Kubernetes Per New Report," Security Boulevard, March 11, 2020.

Contrast SECURITY

Contrast Security provides the industry's most modern and comprehensive Application Security Platform, removing security roadblocks inefficiencies and empowering enterprises to write and release secure application code faster. Embedding code analysis and attack prevention directly into software with instrumentation, the Contrast platform automatically detects vulnerabilities while developers write code, eliminates false positives, and provides context-specific how-to-fix guidance for easy and fast vulnerability remediation. Doing so enables application and development teams to collaborate more effectively and to innovate faster while accelerating digital transformation initiatives. This is why a growing number of the world's largest private and public sector organizations rely on Contrast to secure their applications in development and extend protection in production.

240 3rd Street
2nd Floor
Los Altos, CA 94022
Phone: 888.371.1333
Fax: 650.397.4133

Contrast SECURITY

contrastsecurity.com