SYNOPSYS®

4 Strategies for Securing Container Deployments

Table of contents

Introduction

Containers are popular among organizations transforming their IT operations from physical, single-tenant computing resources to a more efficient, virtual, multitenant infrastructure. The container framework popularized by Docker simplifies and accelerates application deployment by packaging operating system components, applications, and all dependencies into layers within what's known as a container image.

A primary goal of any organization adopting a new technology is to reduce exploitable vulnerabilities and risk. Organizations hesitant to adopt containers are often wary of the challenges of securing containers in production. For their many benefits, containers also represent a new layer in the application stack, which requires a new way of thinking about application security. In its Application Container Security Guide, NIST points out that as containers revolutionize application deployment, organizations must adapt their security strategies to new, dynamic production environments.

Challenges of securing containerized production environments

Just as traditional applications are vulnerable to attack, containerized applications and the containers holding them are as well. Organizations can begin designing an effective container security strategy by understanding the risks that containerization may pose. Here are just a few considerations:

Isolation

Container isolation differs from virtual machine (VM) isolation. The isolation provided by hypervisors in VM systems limits the ability of an attacker to move laterally within an application stack if it is breached. But container applications don't require hypervisors; instead, they share elements of the host operating system. Some organizations worry that if they used containers, a breach would expose more of their sensitive data than it would if they used VMs, which may limit the reach of an attacker.

Runtime complexities

The dynamic nature of containers introduces new runtime complexities that application deployment teams must understand and manage. Applications in containers can make calls to the host to request access to resources, including files, on shared storage systems. If attackers compromise a containerized application, they might gain access to sensitive information on these shared systems. For this reason, IT operations and security teams should monitor their containers' behavior and prevent unauthorized activities.

Vulnerability management

Most container images are created from base images, which are essentially limited, lightweight operating systems. Application container images combine base images with application-specific elements, such as frameworks, runtimes, and the applications themselves. Each layer in a container image is an attack surface that can harbor software vulnerabilities, thereby introducing risk into the organization. But discovering where these risks exist can be difficult, considering some clusters have reached the scale of 10,000 images or more. And even after an organization has scanned all its containers, it must continue to monitor them for newly discovered vulnerabilities in any layer.

Each layer in a container image is an attack surface that can harbor software vulnerabilities.

Strategies and technologies for container security

While securing container clusters may seem daunting, security teams hoping to protect passwords, customer data, personal information, and other sensitive information can—and should—control the security risks associated with containers.

With the right tools, practices, and strategies, organizations can address the challenges of container security described above and protect their containerized applications from attacks. There is no golden goose for container security, so organizations should use a combination of techniques and solutions suited to their IT governance requirements. Below are some common approaches to container security, as well as their pros and cons.

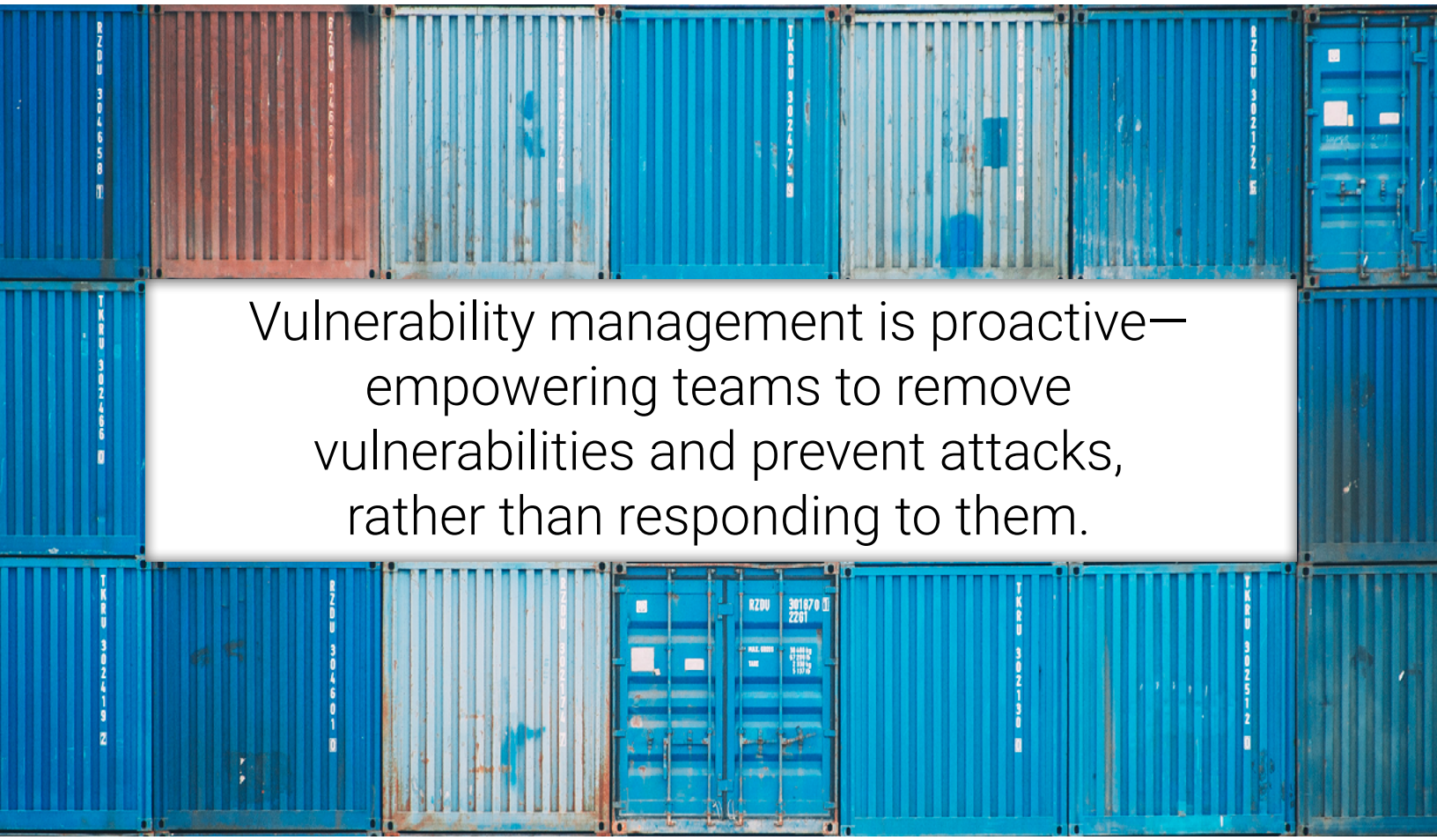
Strategy 1: Conduct manual reviews

According to a study by Forrester, 43% of container users perform regular security audits of their clusters. These security audits may consist of tracking components with known vulnerabilities on spreadsheets or manually testing configurations. Often, an organization will conduct a manual review when it's experimenting with containers. It takes time to determine which processes and technologies are appropriate for a container environment, which is why manual processes work for small, immature deployments.

However, as organizations move more of their container applications into production, this approach does not scale. NIST points out the importance of having dedicated security solutions designed to scale up and down with container clusters. Traditional IT security methods and technologies that are not meant for containerized production environments may leave gaps in application security initiatives.

Strategy 2: Run containers on virtual machines

Another benefit of containers is that they can run anywhere, including within the technology they are disrupting: VMs. Some organizations run containerized applications on VMs to isolate their containers using hypervisors. They do so to prevent attackers from moving laterally within the application stack to access data belonging to other applications, as described earlier. While this strategy can limit the severity of an attack, it will not prevent the attack from happening in the first place.



Vulnerability management is proactive—empowering teams to remove vulnerabilities and prevent attacks, rather than responding to them.

Strategy 3: Employ container runtime security

Runtime security solutions are popular options for organizations hoping to detect and block malicious activity in their running containers in real time. By monitoring network calls to the host and attempts to log into containers, these solutions build behavioral models of every application in an environment. These behavioral models learn which network actions and file system and operating system activities and capabilities to expect.

Whenever runtime security solutions detect that a container has been asked to perform an unexpected function, they can block the action and notify IT teams. And since network security is a runtime responsibility, these solutions can also block blacklisted IP addresses—permitting only legitimate connections. By limiting network access in the face of unexpected network traffic, runtime solutions can shut down a hacker's container network activity in the case of a breach. This limits the extent to which the attacker can move laterally through the container cluster—reducing the "blast radius" of an attack.

Runtime security is an important element of a container security strategy, acting as a last line of defense against malicious actors. However, this approach is reactive rather than proactive.

Strategy 4: Enact vulnerability management

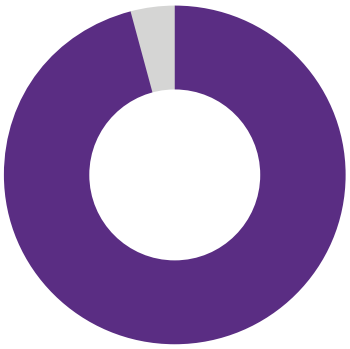
Governance regulations increasingly require a level of continuous monitoring for vulnerabilities in deployed applications. Organizations should take action to prevent attacks by eliminating any latent vulnerabilities that might enable attacks. In contrast to runtime security, vulnerability management is a proactive stance to container security—empowering teams to remove vulnerabilities and prevent attacks before they happen, rather than responding to them.

To secure their containers, organizations must know what they contain. After all, it's not possible to patch something if you don't know it exists. But the widespread use of open source poses a challenge. Open source components appear throughout container images—from the base image to the application layer.

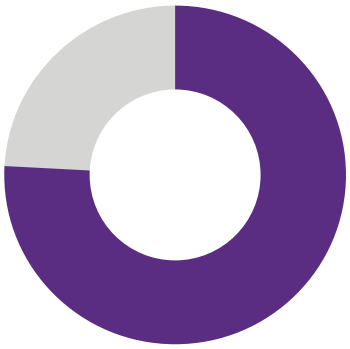
It's not possible to patch something if you don't know it exists.

According to the Open Source Security and Risk Analysis (OSSRA) report, open source components are found in 96% of audited codebases, with the average codebase made of 76% open source. Security risk is prevalent in open source across all industries represented in the report: 84% of codebases contained at least one security vulnerability, and 48% of the codebases contained at least one high-risk vulnerability. Given the pervasive use of open source and the growing scale of container clusters, it's unrealistic to expect organizations to track open source components and their associated vulnerabilities manually.

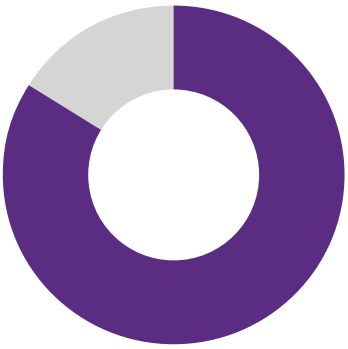
Vulnerability management technology can help organizations keep track of the open source components in container images, as well as their vulnerabilities. While many CISOs and heads of IT departments are wary of the risk that containerization introduces, they can significantly lower that risk by using software solutions that give them continuous visibility into the vulnerabilities in their clusters.



Black Duck On-Demand audits found open source components in **96%** of the applications scanned, with an average **595** components per application.



The average percentage of codebases that was open source was **76%**. Most applications now contain more open source than proprietary code.



84% of the codebases examined contained at least one vulnerability, with an average of **149** vulnerabilities per codebase.

How to choose the best container security strategy for your organization

As container clusters grow in production environments, security processes must scale with them. To get a full picture of the risks in a container cluster, organizations must automate the process of identifying risks and reducing them.

While no single solution will completely secure container clusters, organizations can use different techniques to address some of the risks posed by containerization. Container runtime security solutions can help teams monitor and prevent unauthorized calls to the host, limiting the scope of breaches. This approach can help teams react to attacks in real time. For those interested in proactively reducing risk to prevent attacks, vulnerability management solutions can help them automatically find vulnerabilities and remove them from their clusters—enabling them to reduce the risk of attacks at scale.

Vulnerability management is proactive—empowering teams to remove vulnerabilities and prevent attacks, rather than responding to them.

How Synopsys can help

Black Duck Software Composition Analysis (SCA) integrates directly with orchestration platforms, such as Docker Swarm and Kubernetes, to give IT Operations, development, and security teams automated visibility into, and control of, the open source risks in their container clusters. By automatically scanning images, using source code, file system, and binary analysis, Black Duck discovers, and continuously monitors for, container vulnerabilities at scale.

We can help you find out what's really inside your containers.

Learn more

SYNOPSYS®

The Synopsys difference

Synopsys provides integrated solutions that transform the way you build and deliver software, accelerating innovation while addressing business risk. With Synopsys, your developers can secure code as fast as they write it. Your development and DevSecOps teams can automate testing within development pipelines without compromising velocity. And your security teams can proactively manage risk and focus remediation efforts on what matters most to your organization. Our unmatched expertise helps you plan and execute any security initiative. Only Synopsys offers everything you need to build trust in your software.

For more information, go to www.synopsys.com/software.

Synopsys, Inc.
690 E Middlefield Road
Mountain View, CA 94043 USA

Contact us:
U.S. Sales: 800.873.8193
International Sales: +1 415.321.5237
Email: sig-info@synopsys.com