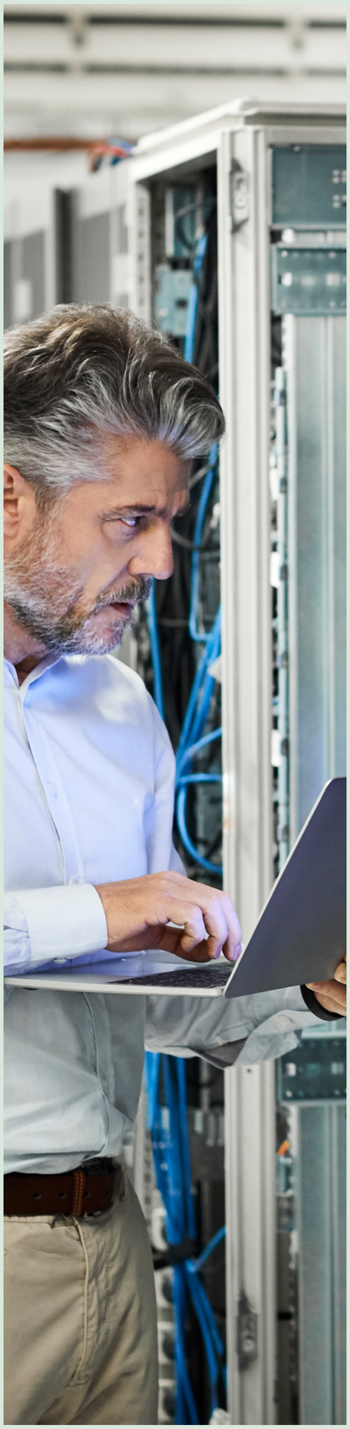
WHITE PAPER



FORTIFYING CONTAINERS WITH INFOSYS SECURITY TESTING SERVICES

Abstract

Containerization has revolutionized the way software applications are deployed and managed, providing unparalleled agility, scalability, and portability. However, the rapid adoption of containers has introduced new security challenges that must be addressed to ensure application and data security.

This paper aims to explore the intricacies of container security. It provides valuable insights on best practices to mitigate risks associated with containerized environments. The paper also delves into the different software development lifecycle stages of container security along with a range of aspects including threat modeling and container runtime security measures with Infosys Security Testing Services.

Infosys®
Navigate your next

Introduction

Over the past two decades, containerized applications have evolved rapidly, changing the dynamics of modern IT infrastructure. Organizations have tested containers and eagerly adopted these extensively to accelerate software deployment, scale seamlessly, and drive digital transformation. The two main drivers of the growing use of containers are ease of maintenance and cost-effectiveness. According to a leading industry analyst report, by 2025, 85% of global organizations will be running containerized applications in production.

Need for Container Security

Containers have vulnerabilities that, when exploited, can adversely impact organizations and their operations. Here are some recent cybersecurity incidents involving containers:

In 2019, Docker Hub suffered a major data breach where hackers gained unrestricted access to the Docker Hub database. The breach affected 190,000 users.

In 2022, Docker Engine honey pots were compromised, targeting Russian and Belarusian websites, resulting in a denial-of-service attack.

In 2022, researchers found that nearly 900,000 misconfigured Kubernetes instances were exposed on the Internet.

Such cybersecurity incidents are on the rise, resulting in financial implications and negative brand perception. A survey by Aqua Security found that 85% of organizations had deployed containers in production environments, and 44% had experienced at least one container security incident in the past year. It also estimated that the average cost of a container security incident was US $250,000. Further, 39% of incidents resulted in lost revenue and 34% led to customer attrition. Another study by Cybersecurity Ventures predicts that global cybercrime damages will more than triple to US $10.5 trillion annually by 2025, up from US $3 trillion in 2015.

As organizations increasingly implement containers, it is imperative that they also focus on container security and reliability. Safeguarding containers is paramount to protect enterprises against cybercrime damages, reduce the cost of security incidents, and maintain customer trust.

Challenges in Container Security Deployments

Despite its rapid growth, containerization is still a nascent strategy. The mainstream adoption of application container technology raises several concerns, security being the foremost. Apart from this, the challenges in implementing security deployments in containers are:

Complexity and scale:

Containerized environments can be highly complex and tend to scale rapidly. This makes it difficult to manage security across several containers, orchestrators, and infrastructure components. It requires streamlined security management tools and procedures.

Limited visibility:

Containers operate in isolated environments, making visibility into all activities and communications challenging. This is particularly true when monitoring and logging container activity, threats, and security issues.

Container runtime security:

Securing the container runtime environment, kernel, and container engine is important. Hackers can compromise the entire container infrastructure by taking advantage of flaws in container runtimes. To reduce risk, regular updates, patching, and hardening measures are required.

Container orchestration security:

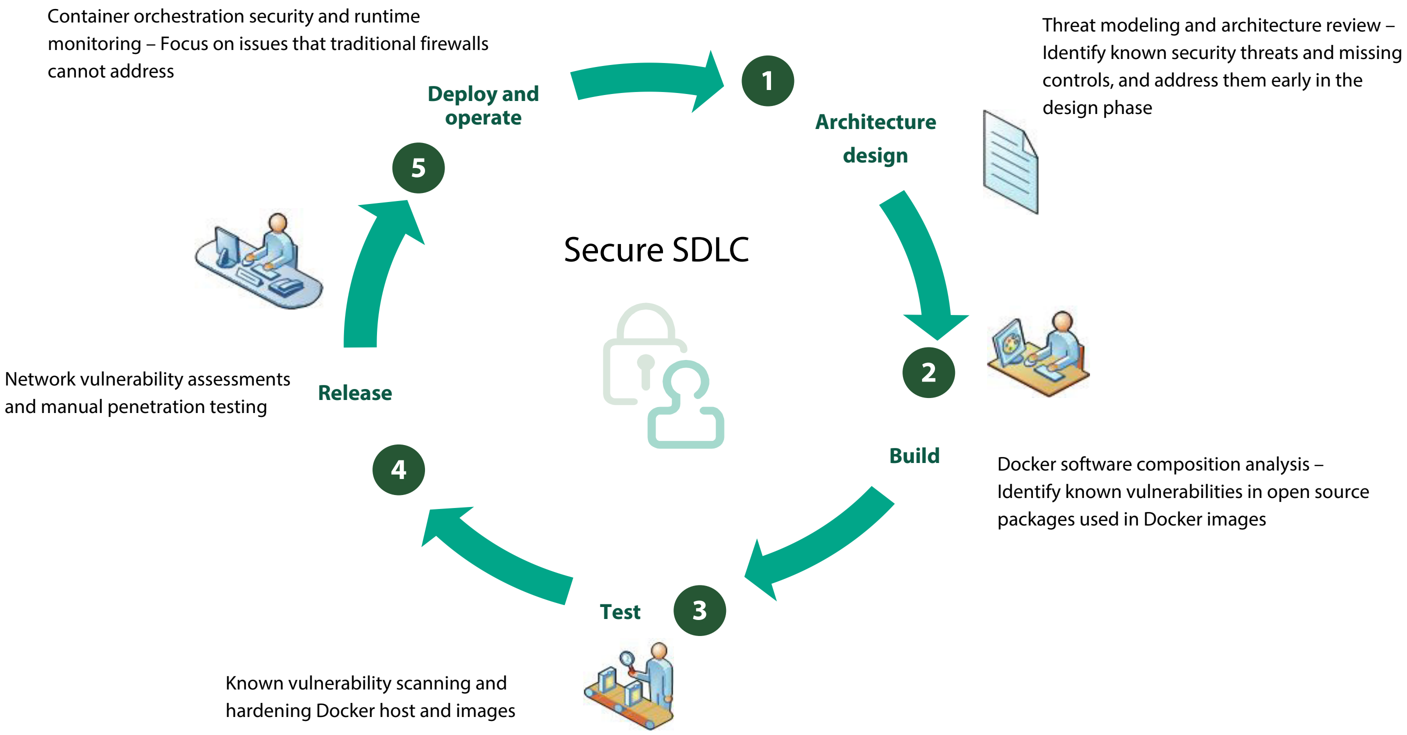
Container orchestration platforms like Kubernetes introduce additional security challenges. These require complex security tasks such as maintaining access restrictions, creating network segmentation, configuring and protecting the orchestrator architecture, and ensuring adequate authentication and permission.

DevOps integration:

Automated CI/CD pipelines, which are frequently used to deploy containers, must be integrated with security procedures. It may be challenging to achieve a balance between security practices and controls, and the need for speed and agility in development.

Implementation of Container Security in the SDLC

Embedding different types of container security testing in each stage of the modern **software development lifecycle** (SDLC) can effectively help overcome the above challenges.



Threat modeling and architecture review – Identify known security threats and missing controls, and address them early in the design phase

Docker software composition analysis – Identify known vulnerabilities in open source packages used in Docker images

Known vulnerability scanning and hardening Docker host and images

Network vulnerability assessments and manual penetration testing

Container orchestration security and runtime monitoring – Focus on issues that traditional firewalls cannot address

Figure 1: Embedding different types of container security testing in each stage of the SDLC

Container security is an ongoing process. Emerging threats make it essential to periodically assess and update security measures and align these with ever-changing security best practices.

Here are some key focus areas to ensure container security:

Known vulnerabilities in container/docker images:

Container images often include various software components, libraries, and dependencies. All of these can have vulnerabilities that attackers can exploit. Regular vulnerability scanning and patching is necessary to mitigate these risks and ensure that the containers are running on secure software stacks.

Container orchestration complexity:

The use of container orchestration platforms like Kubernetes introduce additional complexities and potential security risks. Inadequate security configurations, insecure container orchestrator APIs, and weak authentication mechanisms can expose containers to unauthorized access or malicious activities.

Privilege escalation:

Containers may require elevated privileges or permissions to perform certain tasks. Without proper management, it can lead to privilege escalation attacks where an attacker gains unauthorized access to sensitive resources or executes malicious activities within the container.

Network security:

Containers communicate with other containers and external systems, creating a network attack surface. Without proper network security, attackers can intercept or manipulate container traffic, leading to data breaches or unauthorized access.

Need for automation:

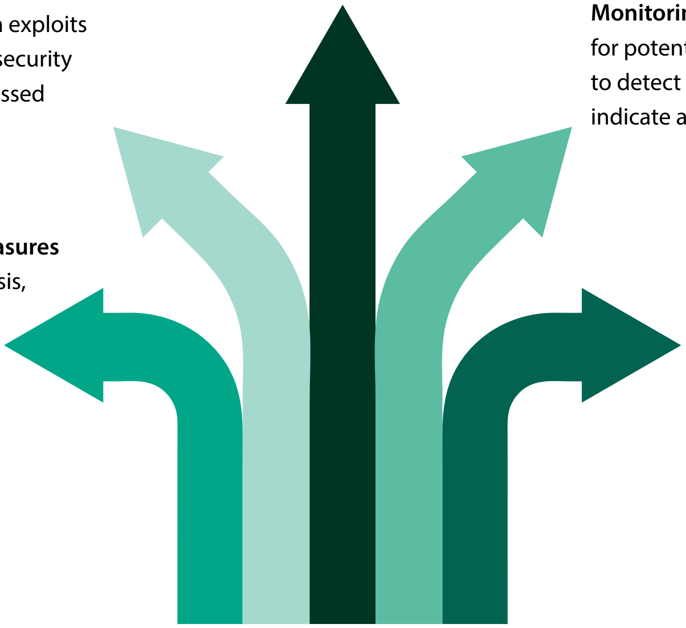
Implementing container security controls in DevSecOps is different from a traditional SDLC. Container security calls for incremental, continuous, and automated quality engineering.

Container Security with Infosys Security Testing Services

Infosys Security Testing provides quality assurance services geared towards digital technologies such as cloud, Internet of Things (IoT), and SAP S/4HANA. Shift-left and DevSecOps are two cutting-edge methodologies that Infosys uses for static and dynamic application security testing.

Infosys leverages DevSecOps to integrate security procedures and controls in the entire development process. The DevSecOps container security approach bridges the gap between development and security by infusing security practices and controls throughout the SDLC. It promotes collaboration, automation, and proactive security measures, ensuring that containers are developed and deployed with a security-first approach.

Some of the key Infosys Security Testing activities for container security include:



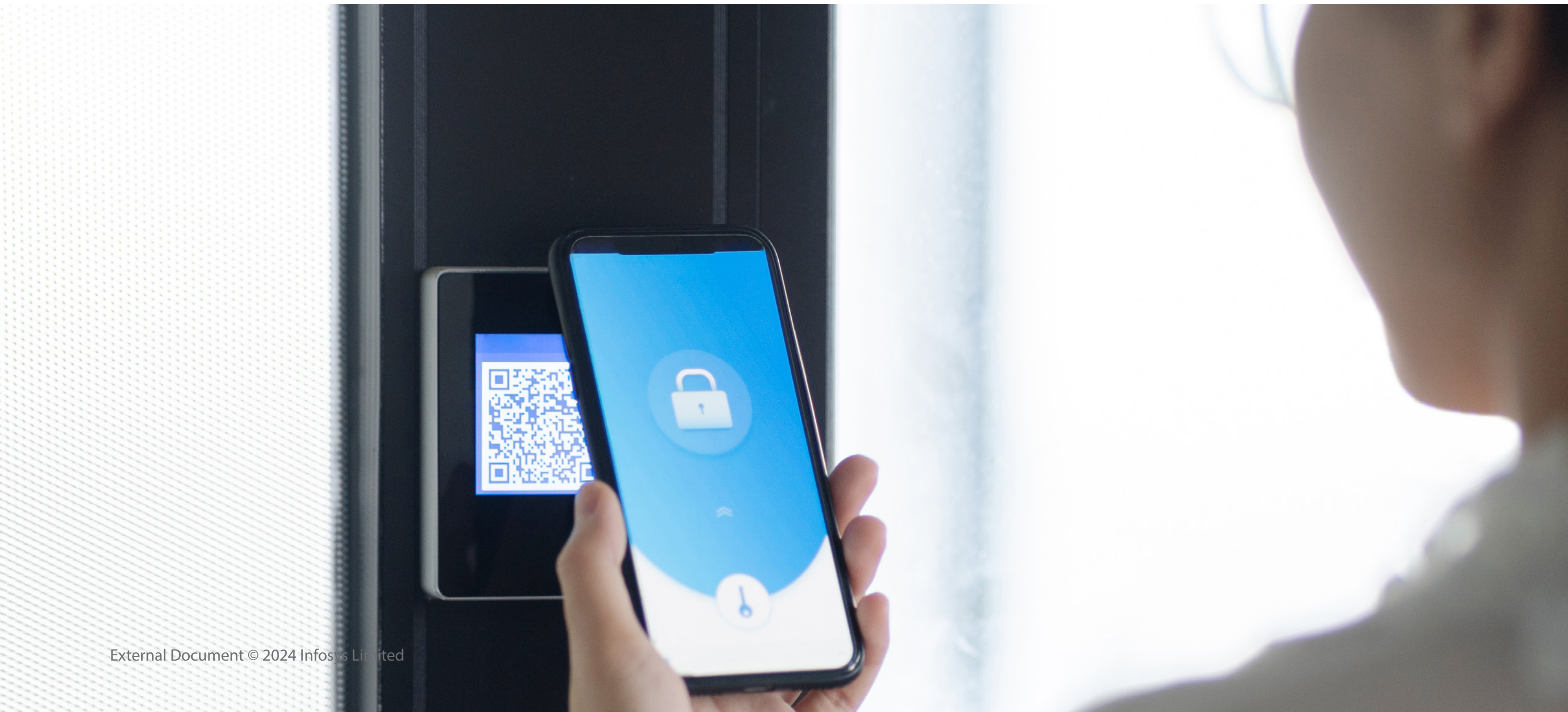
Configuring containers to minimize attack surfaces and reduce the risk of misconfiguration. This involves implementing strict access controls and limiting the number of services and ports exposed to the network

Scanning container images for vulnerabilities and known exploits to ensure that any potential security risks are identified and addressed before deployment

Monitoring containers for potential threats and anomalous behavior to detect any suspicious activity that may indicate a security breach

Implementing security testing measures such as software composition analysis, known vulnerability scanning, and continuous monitoring to identify and address security issues in real-time

Applying security patches and updates to containers in a timely manner to ensure that any known vulnerabilities are promptly addressed



Best Practices for Container Security Implementation

Infosys follows industry-standard best practices for container security. These are recommended for all organizations looking to enhance the security of their containerized environments. These include the following.

Use trusted base images

Start with basic, official, and trustworthy images from reliable sources. Update and patch these base images regularly to mitigate known vulnerabilities.

Ensure image integrity and provenance

Image integrity can be secured by establishing a secure image registry, image signature and verification mechanisms, and enforcing policies for image provenance.

Implement least privilege

Apply the 'least privilege' principle to container settings to limit the access rights and permissions required by containers. To reduce the probability of container compromise, restrict container capabilities and avoid executing containers with root access.

Implement Center for Internet Security (CIS) benchmarks

By leveraging CIS benchmarks, organizations can adopt industry-recognized best practices and enhance the security posture of their containerized environments.

Secure container orchestration platforms

Adhere to the security best practices of the chosen container orchestration platform, such as Kubernetes. This includes securing the control plane, enabling role-based access control, restricting access to the API server, and implementing network policies.

Hardening hosts

Hardening the host operating system is an important aspect of container security as it provides a secure underlying infrastructure for running containers.

When selecting container security testing tools, it is important to consider the specific security needs and requirements of the organization. Vulnerability scanning, runtime protection, access controls, integration, and ease of use are some key selection criteria. Finally, it is worthwhile to evaluate multiple tools and to ensure that the selected tools integrate well with existing technology stack of the enterprise.

Conclusion

Containers have become a widely adopted technology for deploying applications owing to their lightweight nature and ease of movement across various environments. However, their usage creates new security challenges that must be addressed. These challenges include vulnerabilities in the container image, misconfigurations, and potential attacks on the container runtime environment. Acknowledging and mitigating these risks is vital to ensure the security and integrity of applications and their data. Robust security testing measures and best practices help safeguard against potential threats. Infosys Security Testing offers a gamut of services that fortify container security, thereby protecting organizations from potential financial and reputational losses. It helps improve the security posture so organizations can confidently adopt new technologies and meet business goals with agility, speed, and scale.

About the Authors

Kedar J Mankar

Kedar is a Global Delivery Lead for cybersecurity testing. He has extensive experience across various software testing domains. He has led large-scale innovative delivery and transformation programs for Fortune 500 organizations. Kedar has significant collaborative experience, working with teams in security, data, automation, and DevSecOps across multiple geographies and verticals.

Bharat Kumar Rayudu

Bharat Kumar Rayudu is a senior technology architect at Infosys. He possesses vast experience in working and handling teams in security testing, vulnerability assessment and penetration testing, SAP security, DevSecOps, IDAM cloud and container security across multiple geographies and verticals. He has led large size delivery and transformation programs for global Fortune 500 customers and delivered value through different COEs with innovation at core.

For more information, contact askus@infosys.com

Infosys®
Navigate your next