orca security

5 Requirements

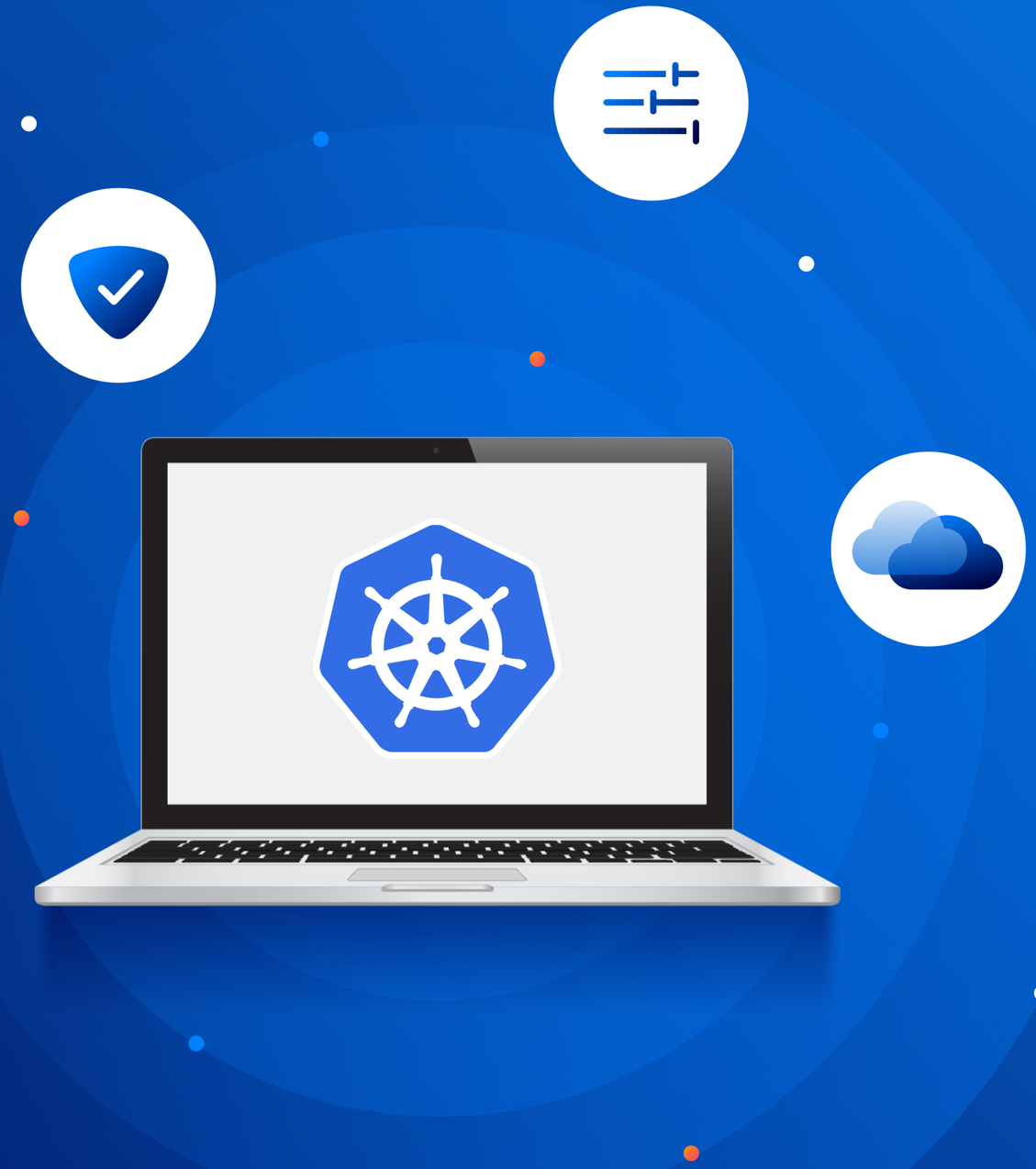
for Integrating Security Across the Full Application Lifecycle

Introduction

Cloud-native applications, built on architectures like containers and Kubernetes, are composed of many interconnected microservices that run on cloud infrastructure — and these architectures are continuing to rise in adoption.

According to the Cloud Native Computing Foundation (CNCF), **96% of organizations use or evaluate Kubernetes.** This adoption is driven by the many advantages that Kubernetes brings, including shorter release timeframes, lower IT costs, increased scalability, and flexibility in multi-cloud environments. Development and DevOps teams are flocking to Kubernetes to take advantage of effortless deployments of their applications, from daily to weekly schedules, to deliver as much value as possible for their end-users.

In addition to container and Kubernetes adoption, enterprises are leveraging Infrastructure-as-Code (IaC) as a technology and process to quickly and repeatedly provision cloud infrastructure. Technologies like AWS CloudFormation templates, HashiCorp Terraform templates, and Kubernetes YAML files can be shared across development and DevOps teams. This allows teams to provide cloud infrastructure repeatedly without writing long, tedious files from scratch.



Why Do Security Teams Need to 'Shift Left'?

However, even though cloud-native technologies are enabling developers to move at the speed of the cloud, if security teams must still test and QA applications before they can be deployed, this can significantly slow down the process. With security being an essential component of all applications, how can organizations ensure development speed while at the same time adhering to security and compliance requirements?

This is where Shift Left comes in. Shifting security left allows organizations to address security issues at an early stage in the software development lifecycle. This approach focuses on testing and fixing issues when development costs are the lowest – before code and applications are deployed in production. Shifting left reduces the risks and costs associated with fixing production security problems.

Cloud-native applications need security that is robust yet as agile as the cloud, instead of security that gets bolted on after applications are deployed. Security needs to be seamlessly integrated into the CI/CD development process through the use of tool integrations, automation capabilities, and APIs.

"By integrating vulnerabilities, context and relationships across the development life cycle, excessive risk can be surfaced, enabling development teams and product owners to focus on remediating the areas of the application that represent the most risk."

Gartner Inc., "Innovation Insight for Cloud-Native Application Protection Platforms", Neil MacDonald and Charlie Winckless, August 25, 2021

How a Cloud Security Platform Can Address Risk Holistically

A single cloud security platform—particularly one that combines data from the cloud infrastructure plane, production workloads, with development security capabilities into one platform—reduces complexity and offers benefits from contextual insights to end-users. Instead of siloed views, cloud native security solutions need to have full coverage and visibility into cloud estates, detect risks across the full technological stack, and use context and relationships to recognize how seemingly unrelated low severity risks can be combined to create dangerous attack vectors.

One of the most important factors in choosing the right cloud native shift left security solution is how well it supports current workflows and is able to detect risk throughout the entire development lifecycle. When considering cloud native security solutions, look for one that provides the ability to build security into the CI/CD process, allowing you to shift left and discover risks as early as possible. The shift left approach helps security leaders reduce friction between cloud development and security teams by encouraging collaboration both before and during production, while reducing the number of tools in the security stack.

In this eBook, we outline the 5 important requirements to consider for integrating security across the full development lifecycle and production cloud estates.

> "Securing cloud-native applications offers enterprises the opportunity to redesign security approaches. Rather than treat development and runtime as separate problems — secured and scanned with a collection of separate tools — enterprises should treat security and compliance as a continuum across development and operations, and seek to consolidate tools where possible."
>
> *Gartner Inc., "Innovation Insight for Cloud-Native Application Protection Platforms", Neil MacDonald and Charlie Winckless, August 25, 2021*

REQUIREMENT #1

Scan your container images and IaC templates

Cloud Security Platforms need to be able to scan container images and Infrastructure as Code (IaC) templates in development environments. Being able to scan these images and templates, wherever they are located, is not a trivial task. Scanning capabilities need to be flexible and support any artifact location, such as the development desktop or CI process.

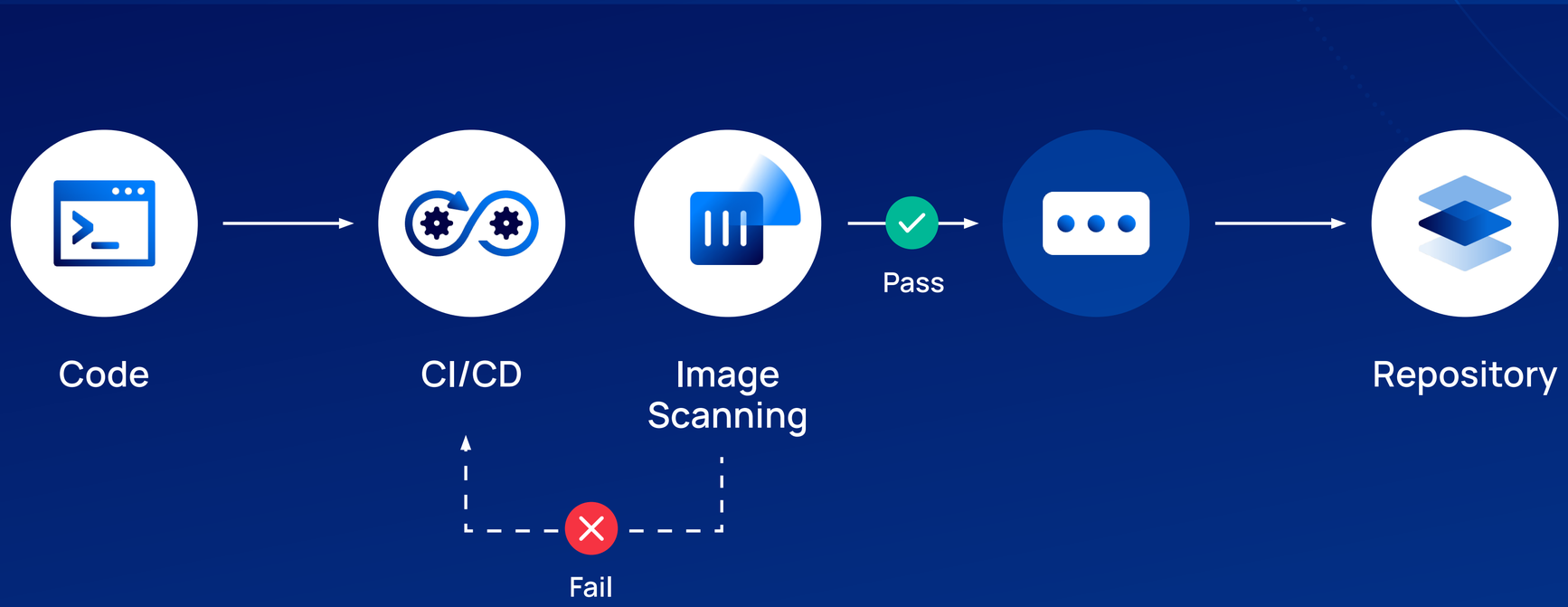
Beyond all of the capabilities mentioned, it's critical to have a cloud solution that provides contextualization for the purpose of risk prioritization. This enables the security and DevOps teams to correlate production risks back to the development image or IaC template that was originally used to create the production instance in order to auto-assign new issues directly to the proper development or DevOps team.

Container image and IaC scanning capabilities should include:

- Vulnerability scanning, to include in every layer of a container image
- Support for security checks specific to containers and IaC solutions, such as HashiCorp Terraform, AWS CloudFormation, and Kubernetes YAML
- Support for CIS benchmark controls and the ability to customize compliance policies
- Ability to detect misconfigurations in both development images and templates
- Results should be available both in the cloud security platform and in developer and DevOps tools
- Seamless integration with common repository managers such as GitHub and GitLab

Container image scanning

Integrated in CI/CD pipelines, preventing unwanted security risks from reaching production



REQUIREMENT #2

Control the build

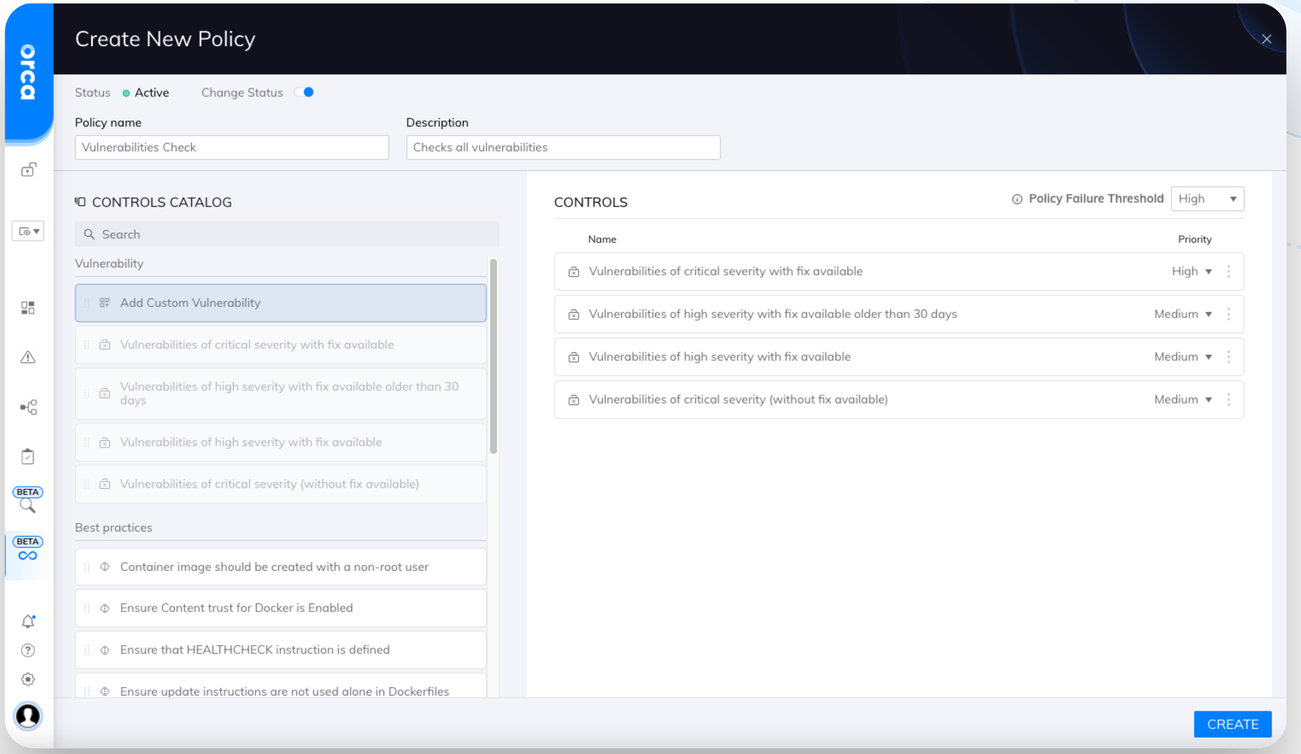
A shift left cloud security solution should allow you to trigger alerts or block builds as part of your workflows. This capability is commonly referred to as guardrails. Security should be well integrated, almost a seamless part of high quality, heavily tested development workflows.

The way guardrails work is by establishing policies for critical issues that if they occur trigger a build to be blocked from being deployed, making it difficult to make large security errors in the deployment process.

Here are some of the guardrail capabilities that the right cloud native shift left security solution can offer you:

- Detect and block critical vulnerabilities and misconfigurations in development images
- Create custom, granular, and flexible policies that can be used to control the guardrail process
- Support APIs or integrations that allow for fine grained control of the build pipeline

The biggest benefit of using guardrails is that it speeds up the deployment process, while ensuring high security quality. With a good guardrail system, developers and DevOps teams can be much more confident in deploying systems without introducing workloads into production that contain serious security risks.



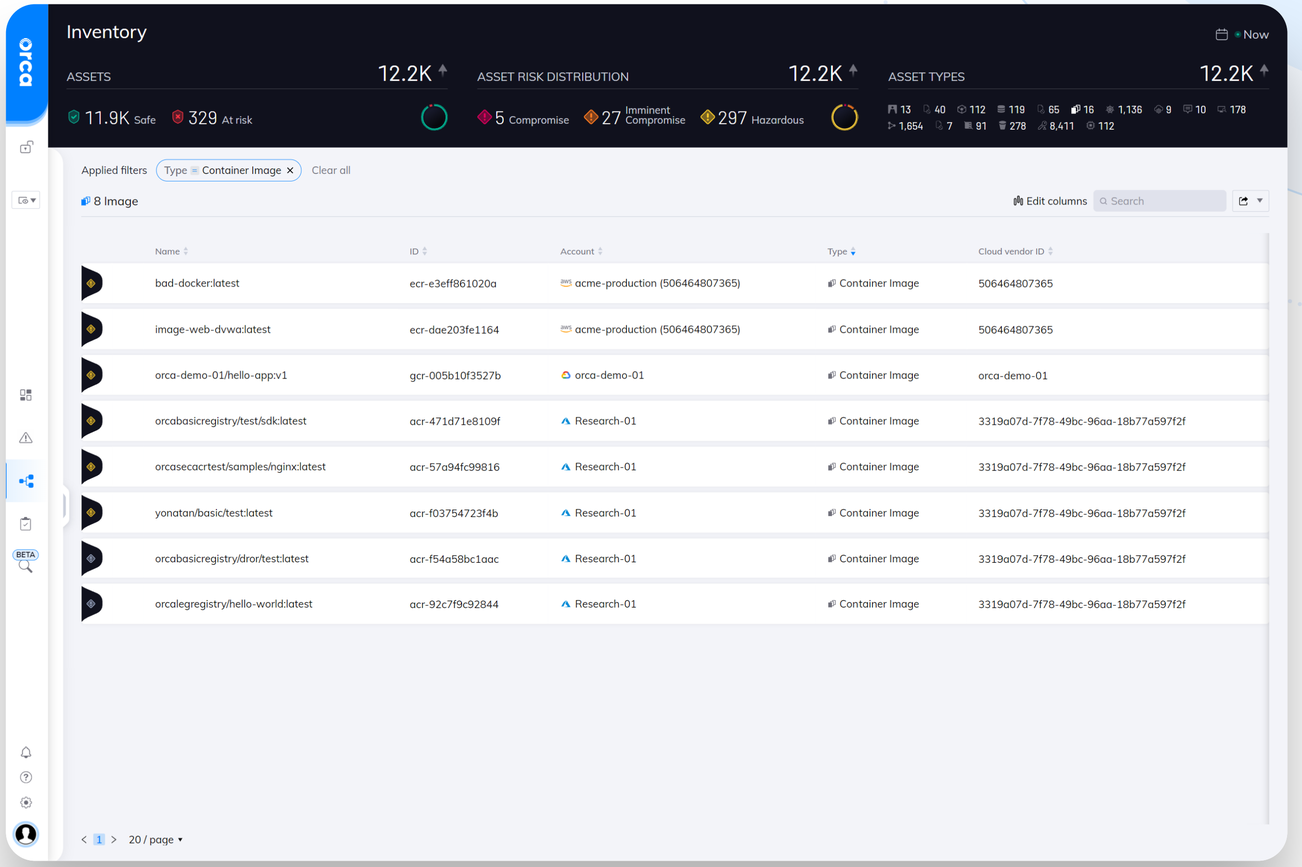
REQUIREMENT #3

Continuously monitor container registries

A container registry is a catalog of images that are used to deploy into runtime environments. Security solutions should be able to continuously monitor common container registries.

Scans of container registries should be performed at regular intervals in order to gain visibility into the quality of artifacts before they reach production. Solutions you'll want to evaluate should be able to support the container registries you use, such as Docker Hub, JFrog, GitHub, GitLab, Google Container Registry, Azure Container Registry, Amazon Elastic Container Registry, etc.

Additionally, security teams should be able to remove stale images and implement policies governing deployments.

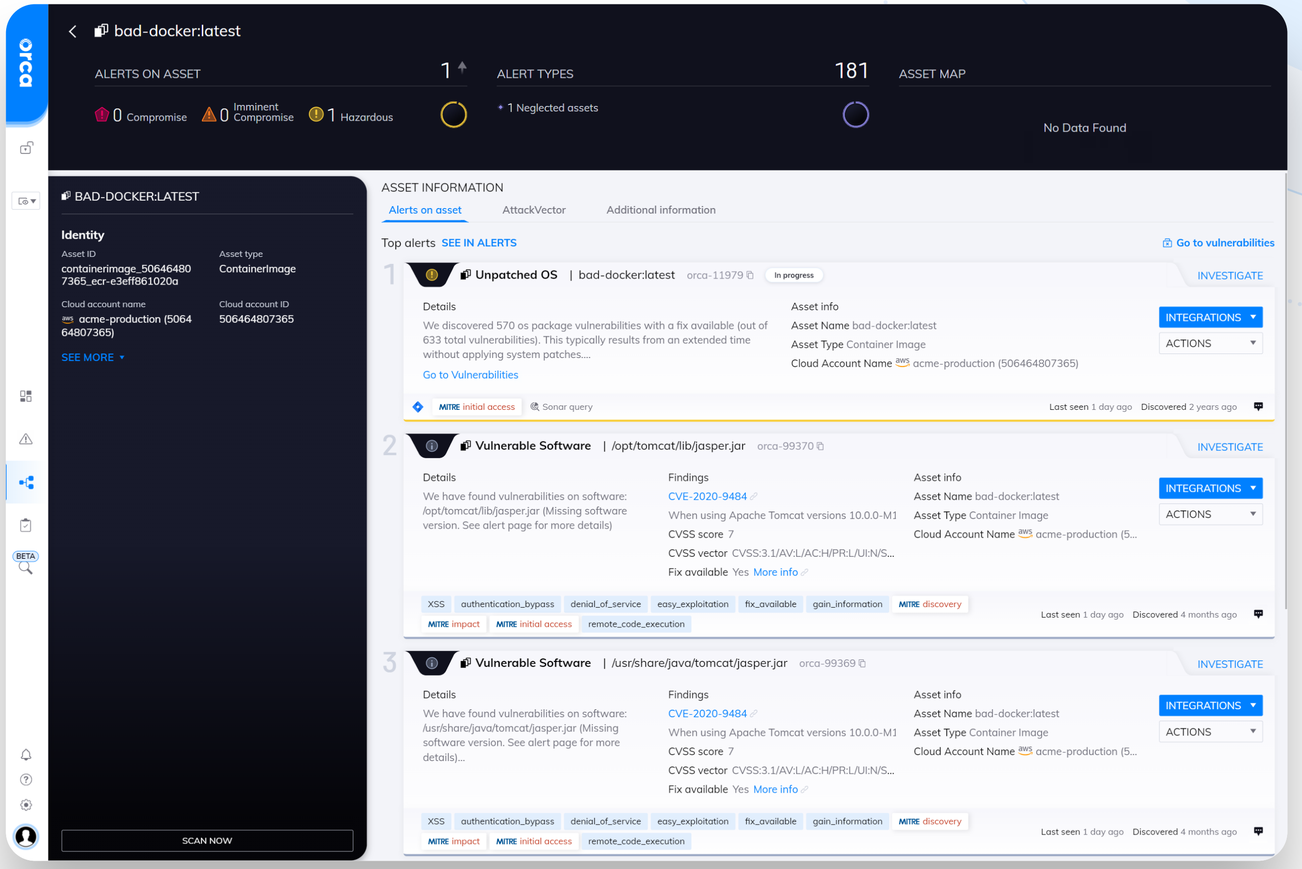


REQUIREMENT #4

Prioritize risks in production

One of the biggest benefits that a cloud native shift left security solution should provide is the ability to view development and production risks holistically in a single, unified data model instead of as a disparate collection of siloed risks. This allows for a highly contextual view of all the different risks in the cloud environment that can then be prioritized based on severity, access, and business impact— allowing security organizations to understand and remediate their most critical issues.

Many tools prioritize vulnerabilities and other risks based only on static scoring systems like CVSS and ignore other relevant contextual factors, such as whether the asset is connected to the Internet or whether it enables lateral movement to sensitive data. As a result, risks are not appropriately prioritized and security teams sometimes spend valuable time working on low risk issues or false positives - when they should be focused on high-risk alerts.



REQUIREMENT #4

When evaluating a cloud native security solution in terms of the ability to prioritize risk, a 'buyer beware' approach is best. Vendors repackaging tools into a single pane of glass with no cohesive integrations between them, may claim, but cannot offer comprehensive visibility across multi-cloud environments. It's not uncommon for vendors to bolt together disparate tool sets (often acquired through acquisition) and rebrand them as a comprehensive, unique security solution.

The other important aspect of prioritization is actually being able to support and detect risks in your existing development workflows. A shift left security solution needs to be able to support technologies and applications that are commonly used in development like Kubernetes, containers, and IaC templates. If you can't scan and detect risk in these development artifacts, you will have gaps in visibility.

When testing options, security leaders need to prioritize the following risks:

Vulnerabilities in VM and container images and IaC templates

Misconfigurations in VM and container images and IaC templates

Malware in VM and container images

Sensitive data in development images

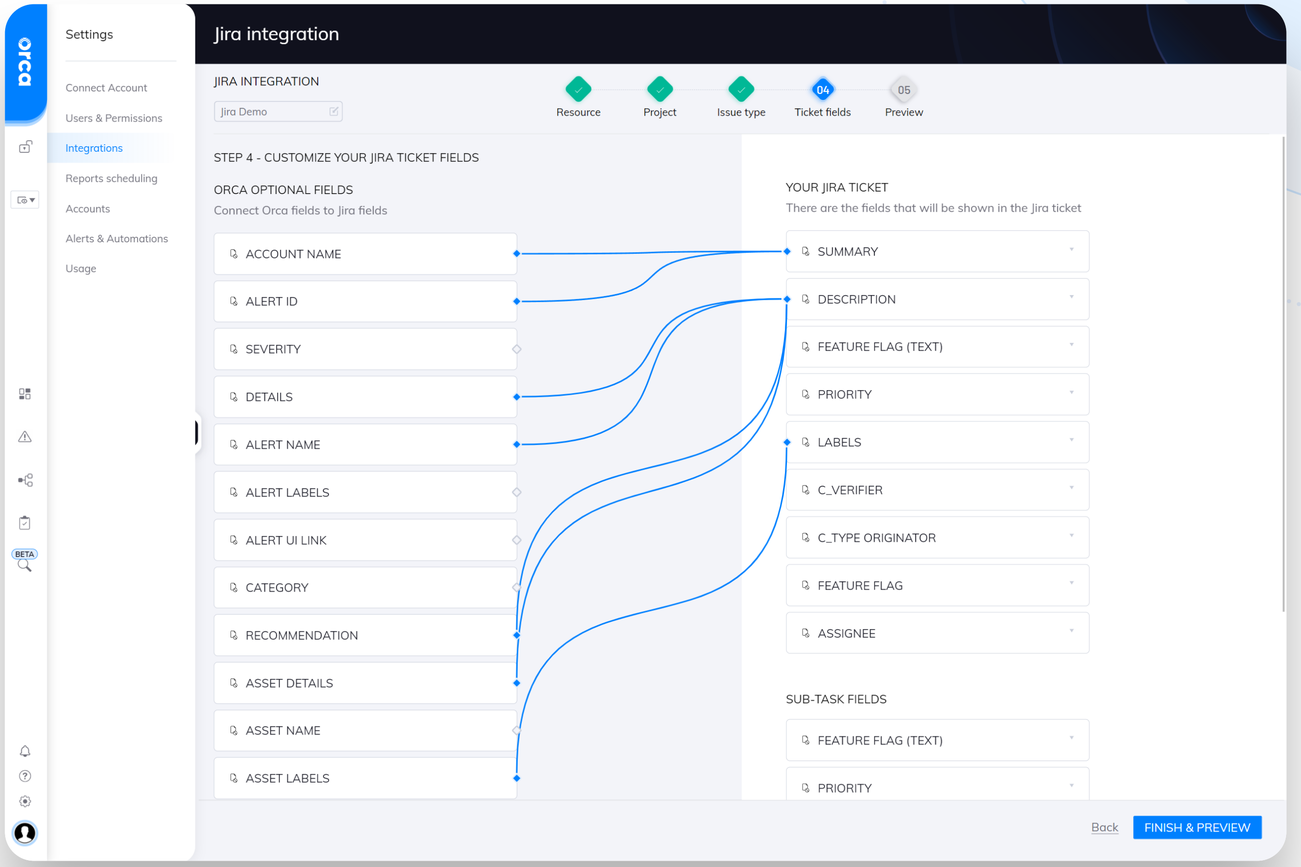
REQUIREMENT #5

Integrate alerts, notifications, and tickets into company -wide workflows

Tight integrations and communication between the different applications and cross-functional teams in your organization is critical. If you can automate the investigation, assignment, mitigation, and remediation of cloud security issues, you'll gain greater efficiency in reducing risk and delivering continuous compliance.

Your cloud native shift left security solution should enable security teams to perform the following actions:

- Forward alerts to email, PagerDuty, OpsGenie, or Slack, and enable automated ticketing with Jira or ServiceNow
- Provide alerts, including rich contextual information, to allow remediation teams to operate independently and efficiently
- Allow you to ingest data into a SIEM for analytics
- Support an API first strategy that facilitates granular control over every aspect of the CI/CD pipeline



Summary

Cloud native security solutions combine cloud workload and control plane intelligence, allowing the holistic insight that you just can't get with separate solutions. These platforms should extend powerful capabilities to the left to support both risk detection and prioritization early on in the development process. With the right technology in place, you can achieve tight integration with development workflows to manage critical risks much earlier and at reduced cost to the bottom line.

Introduce security checks early in the SDLC, preventing security hazards from reaching production, while gaining insight on "what would have happened" if the change was done.

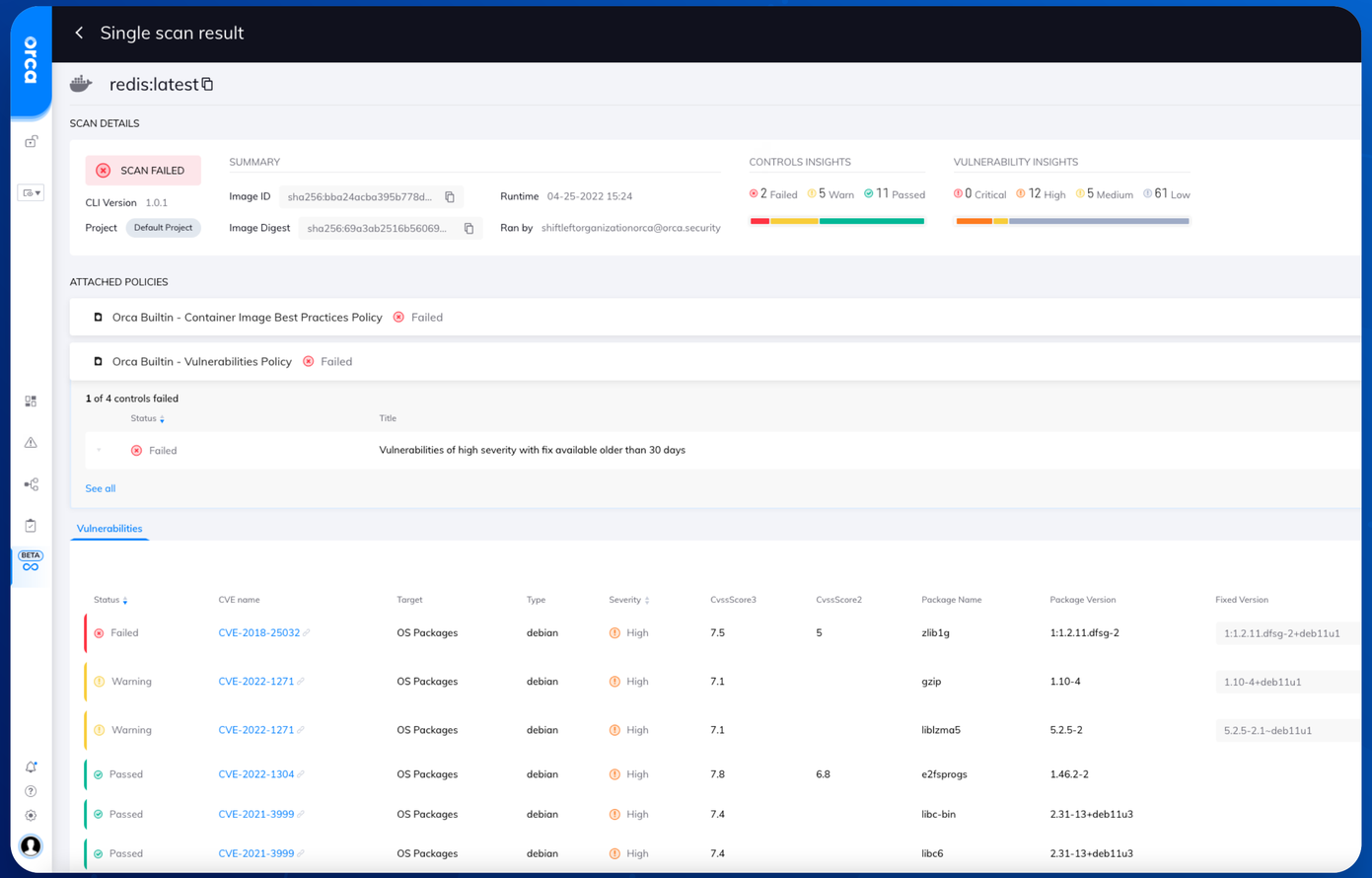
Orca will detect issues in images, registries, and Infrastructure as Code.

The shift left cloud native security solution should be able to provide the five capabilities discussed in this guide:

- Scan and detect risks in container images and IaC templates
- Create granular and customizable guardrails to be able to block builds if needed
- Support continuous monitoring of images in registries
- Prioritize risk in production and development environments
- Integrate alerts, notifications, and tickets into company-wide workflows

Try Shift Left Security Today

Finding a solution that's right for your organization and meets all the requirements suggested in this guide can be challenging. That's why Orca Security invites you to sign up for a free 30-day trial of our cloud security solution with shift left capabilities, without any obligation whatsoever. If the vendor is confident of their platform, it should be easy to test the solution with a free trial or risk assessment. When you test your solution before making a commitment, you can be confident in your ultimate decision. A trusted security provider should be ready with a proof-of-concept to help you see how their technology works in your cloud environment.



About Orca Security

Orca Security provides instant-on security and compliance for AWS, Azure, and GCP - without the gaps in coverage, alert fatigue, and operational costs of agents or sidecars. Simplify cloud security operations with a single CNAPP platform for workload and data protection, cloud security posture management (CSPM), vulnerability management, and compliance. Orca Security prioritizes risk based on the severity of the security issue, its accessibility, and business impact. This helps you focus on the critical alerts that matter most. Orca Security is trusted by global innovators, including Databricks, Autodesk, NCR, Gannett, and Robinhood.

Connect your first account in minutes:
https://orca.security or take the free cloud risk assessment.