VMware Carbon Black Container

Advanced detection and response for securing modern applications

At A Glance

Use cases

- Continuous image scanning from build to run
- Securing containers and Kubernetes applications
- Increase visibility into Kubernetes environments
- Container image malware detection and prevention
- Ransomware and cryptomining protection
- Continuous compliance

Benefits for security teams

- Easier response process and alert triaging
- Alert context and noise reduction
- Visibility into the security posture of Kubernetes
- Prioritized vulnerability reporting
- Easily define and customize security policies
- Enable developers to address vulnerabilities and misconfigurations at build
- Connect image vulnerabilities to specific running workloads

Containers and Kubernetes have become synonymous with the modern application transformation. As the rapid adoption of containers and cloud native practices continues to grow exponentially, so does the attack surface and resulting opportunities for back actors to strike. A new approach to security is required for visibility and control in these highly dynamic, complex environments.

As Security Operations Center (SOC) teams face increasing threats, they're combating additional sets of challenges such as:

- The complexities of cloud native environments
- Containers running in production with limited or no security
- Disparate tools that create coverage gaps in their environment
- Unacceptably high level of false positives
- Legacy detection and prevention controls

To improve the security analyst experience, teams must modernize tools and processes, close visibility gaps, and remove friction from the detection, incident investigation, and response workflow.

A unified security strategy from development to production is critical to detect vulnerabilities and misconfigurations early in application development and minimize the attack surface. VMware Carbon Black Container™ secures cloud native environments throughout the entire software development lifecycle (SDLC) and integrates security into every layer of the application and its environment.

Detect threats faster with Cloud Native Detection and Response

Security teams require visibility into the processes running in cloud native applications, and the ability to enable the same level of protection for container environments that they have for endpoints. Without context for alerts, such as pinpointing which cluster is generating a specific alert, they are not only ineffective but a drain on critical resources.

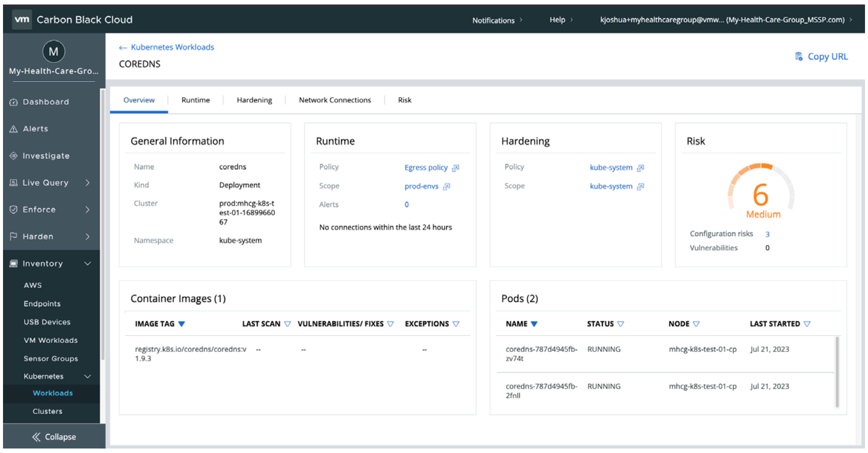
With Carbon Black Container, security teams can enable advanced threat detection for cloud native applications. Security teams benefit from the visibility and context at each layer of the application for easier alert triage and faster remediation. Leveraging custom watchlists, security teams receive actionable alerts for Kubernetes and container risks that integrate seamlessly into existing workflows and operational processes.

Benefits for DevOps teams

- Automated image scanning
- Fast and easy development and setup
- Seamlessly integrate into the CI/CD pipeline and existing processes
- Address vulnerabilities and misconfigurations at build
- Enable speed of delivery without compromising security
- Complete visbility into application connectivity and configuration

Features

- Integrated alert dashboard
- Risk prioritization
- EDR and NGAV for Kubernetes
- Image scanning and detection for vulnerabilities, misconfigurations, malware and secrets
- Network visibility map
- Workload anomaly and threat detection



COREDNS

General Information

Name	coredns
Kind	Deployment
Cluster	prod:mhcg-k8s-test-01-1689966067
Namespace	kube-system

Runtime

Policy	Egress policy
Scope	prod-envs
Alerts	0

No connections within the last 24 hours

Hardening

Policy	kube-system
Scope	kube-system

Risk

Configuration risks	3
Vulnerabilities	0

Container Images (1)

IMAGE TAG	LAST SCAN	VULNERABILITIES/ FIXES	EXCEPTIONS
registry.k8s.io/coredns/coredns:v1.9.3	--	--	--

Pods (2)

NAME	STATUS	NODE	LAST STARTED
coredns-787d4945fb-zv74t	RUNNING	mhcg-k8s-test-01-cp	Jul 21, 2023
coredns-787d4945fb-2fnll	RUNNING	mhcg-k8s-test-01-cp	Jul 21, 2023

Figure 1: Kubernetes workload dashboard in the Carbon Black Cloud console.

Cloud Native Detection and Response capabilities allow security analysts to detect malicious activity inside containers, understand which container or Kubernetes workload and node the alert is coming from, and provide visibility into the steps an attacker took to infiltrate the environment. Due to the ephemeral nature of containers, any previous threat data is no longer available for security teams once that container itself is destroyed. Carbon Black Container records detailed historical event data for containers and Kubernetes to enable security teams to correlate past events and current behavior for more effective threat detection and faster response.

Gain visibility and context into network behavior

Security teams often lack visibility into workloads that are running in a given cluster, and the connectivity to and from those workloads. To simplify this process, the network visibility map allows security teams to view these workload connections in a single map of the application architecture. The network visibility map provides detailed visibility and context to better understand the application architecture and network traffic behavior. Tracking the application's internal and external traffic by customizing group-to-group internal or external IPs can help gain the control required in such a chaotic environment. To get a clean view of an application, filters allow the connectivity of the map to remove unnecessary "noise" such as system namespaces. Security analysts can also use similar filters to better understand what connection is encrypted or not encrypted to gain full visibility into your application traffic posture. The goal of the networking map is to give teams a better understanding of the connectivity and configuration of applications installed in the cluster.

Detect vulnerabilities and malware running in production

The widespread use of third-party image registries when developing applications means that security teams need a way to monitor and scan images to ensure no vulnerable images are running in production. Carbon Black Container continuously scans and monitors all images running in production. Automated continuous monitoring allows for teams to customize alerting for

Platform Support

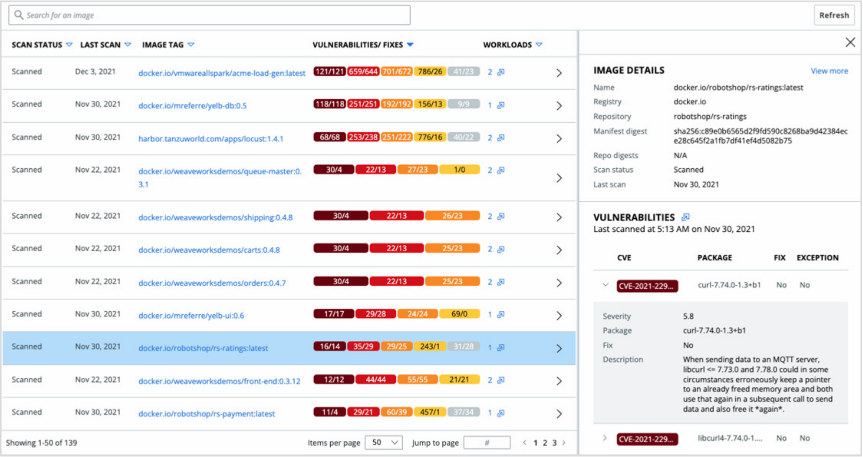
- Upstream Kubernetes
- VMware Tanzu™ Kubernetes Grid™
- RedHat Open-Shift
- Amazon Elastic Kubernetes Service (EKS)
- Google Kubernetes Engine (GKE)
- Azure Kubernetes Service (AKS)

For information on additional platforms supported, please reach out to a VMware representative.

Learn More

For more information or to purchase VMware products, call 877-4-VMWARE (outside North America, +1-650-427-5000), visit vmware.com/products, or search online for an authorized reseller. For detailed product specifications and system requirements, please refer to the product documentation.

potentially vulnerable images or even prevent the images from running.



SCAN STATUS	LAST SCAN	IMAGE TAG	VULNERABILITIES/ FIXES	WORKLOADS
Scanned	Dec 3, 2021	docker.io/vmwareallspark/acme-load-gen:latest	121/121 659/644 701/672 786/26 41/23	2
Scanned	Nov 30, 2021	docker.io/mreferre/yelb-db:0.5	118/118 251/251 192/192 156/13 9/9	1
Scanned	Nov 30, 2021	harbor.tanzuworld.com/apps/locust:1.4.1	68/68 253/238 251/222 776/16 40/22	2
Scanned	Nov 22, 2021	docker.io/weaveworksdemos/queue-master:0.3.1	30/4 22/13 27/23 1/0	2
Scanned	Nov 22, 2021	docker.io/weaveworksdemos/shipping:0.4.8	30/4 22/13 26/23	2
Scanned	Nov 22, 2021	docker.io/weaveworksdemos/carts:0.4.8	30/4 22/13 25/23	2
Scanned	Nov 22, 2021	docker.io/weaveworksdemos/orders:0.4.7	30/4 22/13 25/23	2
Scanned	Nov 30, 2021	docker.io/mreferre/yelb-ui:0.6	17/17 29/28 24/24 69/0	1
Scanned	Nov 30, 2021	docker.io/robotshop/rs-ratings:latest	16/14 35/29 29/25 243/1 31/28	1
Scanned	Nov 22, 2021	docker.io/weaveworksdemos/front-end:0.3.12	12/12 44/44 55/55 21/21	2
Scanned	Nov 30, 2021	docker.io/robotshop/rs-payment:latest	11/4 29/21 60/39 457/1 37/34	1

Showing 1-50 of 139 Items per page 50 Jump to page # ‹ 1 2 3 ›

IMAGE DETAILS View more

Name	docker.io/robotshop/rs-ratings:latest
Registry	docker.io
Repository	robotshop/rs-ratings
Manifest digest	sha256:c89e0b6565d2f9fd590c8268ba9d42384ece28c645f2a1fb7df41ef4d5082b75
Repo digests	N/A
Scan status	Scanned
Last scan	Nov 30, 2021

VULNERABILITIES
Last scanned at 5:13 AM on Nov 30, 2021

CVE	PACKAGE	FIX	EXCEPTION
CVE-2021-229...	curl-7.74.0-1.3+b1	No	No
CVE-2021-229...	libcurl4-7.74.0-1....	No	No

Severity	5.8
Package	curl-7.74.0-1.3+b1
Fix	No
Description	When sending data to an MQTT server, libcurl <= 7.73.0 and 7.78.0 could in some circumstances erroneously keep a pointer to an already freed memory area and both use that again in a subsequent call to send data and also free it *again*.

Figure 2: Container image scanning and vulnerability data with detailed context.

Prevent runtime vulnerabilities and malware

Scanning container images for vulnerabilities, misconfigurations, and malware during development is one of the best ways to reduce risk and exploitability, especially because third-party image registries are often used during development. With Carbon Black Container, DevOps teams can scan images throughout the application lifecycle, both in the CI/CD pipeline and in production. This crucial visibility allows DevOps and Security teams to ensure only approved images which meet your organization's compliance and security standards are being deployed in production. Automated, continuous scanning and monitoring ensures that any images running in production don't become risks.

Enforce security standards, best practices and compliance

Carbon Black Container enables DevSecOps teams to enforce security standards and compliance by restricting container deployments to prevent running unauthorized workloads. They can also configure minimum security standards and compliance requirements to align with security frameworks, such as STIG or CIS, along with Kubernetes best practices. Once in place, these policies can be automated for continuous protection, making proper configuration easier than ever. Leveraging pre-built compliance templates, organizations can further customize to their specific and unique environment and needs. Additionally, custom scopes can be created for different policies. Policy scoping also includes hierachy functionality, that ensure policies don't collide. As a part of the hierarchy, any policy at the namespace level will override the cluster policy.