Continuous Security and Visibility for Red Hat OpenShift

sysdig

Contents

About This Guide

Speed, agility, and scalability are no longer “nice to haves” for IT leaders. These are now core to the way modern organizations operate, and it is critical for CIOs to ensure they have a modern foundation that enables them to quickly move and innovate.

Modern solutions like containers, microservices and hybrid cloud disrupt the way enterprises implement security processes. Containers provide a great level of portability and isolation, which make them ideal for moving applications from development into production. They are like black boxes, however, which means it’s harder to see what’s inside in order to monitor and secure them. As enterprises begin to move from initial sandbox to production deployments, they face operational challenges in maintaining container security and reliability.

Today’s businesses need security that keeps pace with the speed and agility of the cloud and containers but doesn’t slow down the very processes that deliver faster results. This duality in goals – accelerating delivery while ensuring security – demands an approach that both protects data and workloads, and facilitates agile application development.

The Sysdig Secure DevOps Platform provides security to confidently run containers, Kubernetes, and cloud. With Sysdig, you can secure the build, detect and respond to threats, and continuously validate compliance. In addition, our solutions help you maximize performance and availability by monitoring and troubleshooting cloud infrastructure and services.

The Sysdig SaaS platform is radically simple to run and scale and is built on an open-source stack that includes Falco and sysdig OSS, the open standards for runtime threat detection and response.

By creating a secure Devops workflow that integrates security, compliance, and monitoring, organizations can accelerate deployment and confidently run container and cloud workloads in production on Red Hat OpenShift with Sysdig. This allows you to:

- Speed up deployment by validating security policies and configurations during the build process.
- Continuously assess container and infrastructure compliance.
- Stop runtime threats without impacting performance.
- Prevent issues by monitoring performance and health across infrastructure, services, and applications.
- Conduct incident response using detailed records.

This guide offers a framework for establishing comprehensive security for Red Hat OpenShift environments with specific recommendations for how Sysdig can complement and enhance native Red Hat security capabilities. This will help you confidently run containers, Kubernetes, and cloud.

Addressing the unique needs of different Red Hat OpenShift stakeholders

The primary goal of OpenShift is to provide a great experience for development, operations, and security teams to build, deploy, and securely run containerized workloads and accelerate container application deployment. But different teams and roles have different concerns and points of view on what security means, what's required to move into production, and how to implement new security processes.

Developers

OpenShift helps developers take advantage of both containerized applications and orchestration without having to know the underlying infrastructure details. OpenShift pipelines streamline the process of building, distributing, and deploying containerized applications. Using Source-to-Image (S2I), an open-source framework for combining source code and base images, developers can push changes to a repository (such as Github).

OpenShift will create a container image from the source code and push it to a built-in private registry. Ensuring these images are free of known vulnerabilities and following security best practices is a major challenge that often compromises application integrity.

Cloud/DevOps

Cloud and DevOps teams are responsible for maintaining high availability, quality of service, and health of the application and infrastructure. Teams often use the built-in web console to manage the infrastructure and platform capabilities, and also to leverage playbooks to automate application deployments. DevOps teams are required to ensure that they build security into the platform with tools like Falco, the open-source cloud native runtime security project, Pod Security Policies, network policies, and more.

Security and compliance

To be effective at preventing threats, identifying risk, and isolating vulnerabilities, security operations, SecOps, DevSecOps, and CSIRT teams need to continuously monitor OpenShift environments. This protects them against anomalous behavior and zero-day attacks, as well as perform incident response if a violation occurs. Security teams also set policies based on compliance frameworks and internal requirements, and apply those to the various resources operating in the OpenShift environment. In addition, security teams must monitor new container infrastructure and applications that are deployed to ensure they conform with regulatory and internal compliance requirements.

Managing security and visibility on Red Hat OpenShift with Sysdig

With unified security, compliance, and monitoring, you can confidently build and run cloud-native workloads on OpenShift in private, hybrid, and multi-cloud environments. By automating these critical capabilities for a secure DevOps workflow, you can maximize performance, increase agility, optimize data integration across apps and other data repositories, manage security risk, and ship cloud applications faster.

Red Hat provides security capabilities, including:

- Host infrastructure with RHEL/RHCOS.
- Vulnerability scanning with Clair.
- Extensive compliance audit workflows with OpenSCAP.
- Built-in security controls like RBAC and OAuth, Pod Security Context, Security Context Constraints (SCC) and Pod Security Policy to enforce them, Network Policy and Image Policy capabilities.

Sysdig extends OpenShift capabilities, providing additional security and monitoring capabilities to:

- Secure the build pipeline
    - Automate scanning within CI/CD pipelines and registries.
    - Efficiently flag vulnerabilities and identify owners.
    - Block vulnerable images from being deployed.
- Detect and respond to runtime threats
    - See all threats with Falco, the open standard for detection, and implement zero-day threat detection.
    - Prevent lateral movement with Kubernetes network policies.
    - Conduct incident response using detailed records.
    - Get deep runtime visibility into cloud and container services.
- Continuously manage cloud posture and compliance
    - Identify misconfigurations and compliance violations at build and runtime.
    - Monitor account and access security at the individual and group levels.
    - Measure progress with detailed reports.
    - Save time with out-of-the-box policies for PCI, NIST, and SOC2.

- Monitor containers, Kubernetes, and cloud services
  - Prevent issues by monitoring performance and capacity.
  - Accelerate troubleshooting using granular data.
  - Scale Prometheus monitoring across clusters and clouds.
  - Audit container activity and accelerate incident response

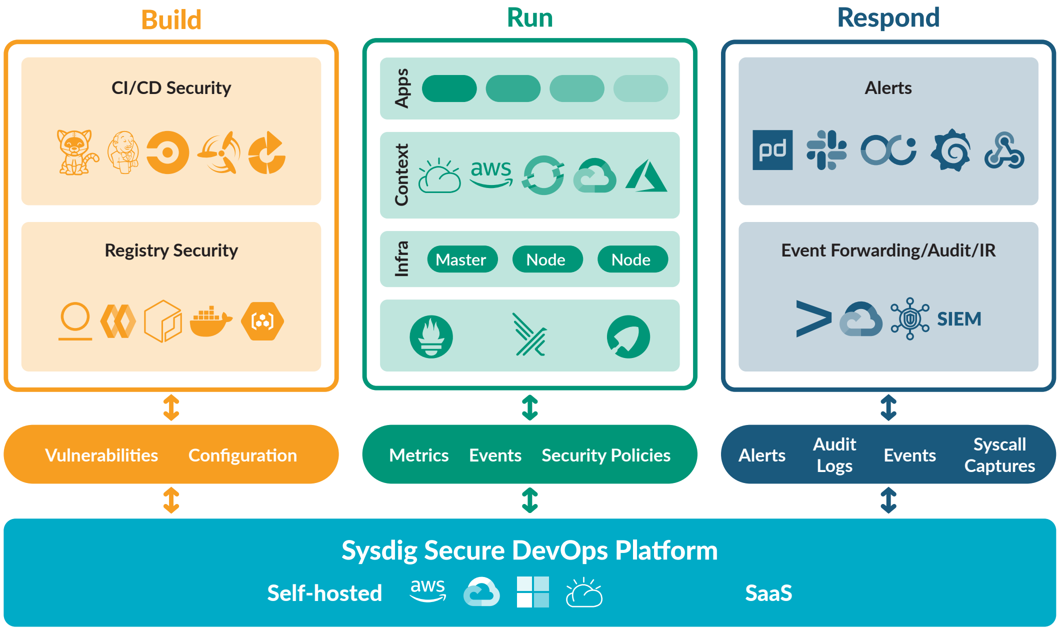
At Sysdig, we provide the only comprehensive, unified platform that features cloud and container security and monitoring. We incorporate the capabilities of a Cloud Workload Protection Platform (CWPP) with Cloud Security Posture Management (CSPM), as well as health and performance observability. We equip DevOps and Security teams with a single source of truth across cloud workloads, accounts, containers, and Kubernetes.

These tools operate as a unified security and visibility layer for OpenShift and cloud environments to eliminate silos of information that exist across operations, development, DevOps, and security teams. We enable security and DevOps teams to accurately identify and triage incidents, quickly determine cause, and perform forensics even for container workloads that are no longer running.

Using the Sysdig platform, security and DevOps teams can report on security issues across their entire OpenShift and cloud environment, including suspicious user behavior, threats to data, and vulnerabilities affecting running images. For example, if a new vulnerability is reported, Sysdig can help your DevOps teams quickly identify the affected images in a particular public cloud region, namespace, cluster, etc., and help you determine the team that owns the fix. With this approach, you can resolve issues quickly by analyzing vulnerabilities and granular system data automatically correlated to both cloud and Kubernetes context.

We help you deliver reliable and secure cloud applications and provide centralized visibility and security for operating containers on OpenShift at scale. With a single agent deployed per cluster node, the Sysdig platform can scale to 10,000+ nodes to secure and monitor your workloads and infrastructure.

You can get started quickly with guided onboarding, out-of-the-box dashboards, and curated workflows. Because Sysdig plugs into your cloud environment and existing DevOps workflow using automation and out-of-the-box integrations, visibility and security controls won’t slow you down.



Sysdig provides container and orchestration insights for OpenShift and cloud using four key technology innovations:

- ImageVision™ identifies and prevents images with vulnerabilities or misconfigurations from being shipped.
- ContainerVision™ gives request-level visibility inside your containers and across microservices. It provides in-depth metrics and events without invasive instrumentation.
- ServiceVision™ integrates with OpenShift to automatically enrich all of your metrics and events with orchestration metadata.
- CloudVision™ enables a consolidated view of cloud activity using cloud logs.

Securing Red Hat OpenShift

Let's look at the various security controls provided by Red Hat and how Sysdig extends security, compliance, and monitoring for OpenShift across the cloud-native stack and container lifecycle.

Host Security

Red Hat Enterprise Linux (RHEL) and CoreOS (RHCOS) come with extended Linux security features, such as:

- Linux namespaces and control groups used by CRI-O, to create the isolated container, limiting visibility and resources of the processes running inside.
- seccomp profiles, that restrict the system calls that a container can execute.
- SELinux profiles to enforce access control policies.
- Fine-grained control over superuser permissions, allowing certain behaviors without running as the root user.
- Controlled immutability (with RHCOS) to lock down management via remote management from the OpenShift cluster and limit OS modifications to only a few system settings.

These critical capabilities ensure a baseline level of trust and security in the host operating systems (RHEL/RHCOS). These are typically applied to containers through the container engine, like CRI-O, and orchestrated by OpenShift through Pod Security Context and Pod Security Policy definitions.

Authentication and Authorization

User access to OpenShift is provided through standard interfaces, including the Web UI, CLI, and APIs. Additionally, services interact with OpenShift so they can gain awareness of their orchestration state and execute actions against the platform. Imagine a CI/CD pipeline pushing a new deployment into production. How do you control and measure who can do what?

Red Hat provides...

The OpenShift Container Platform master includes a built-in OAuth server. Users and service accounts can obtain OAuth access tokens to authenticate themselves to the API as a form of role-based access control (RBAC). Often, OAuth leverages an existing external directory like LDAP or Active Directory.

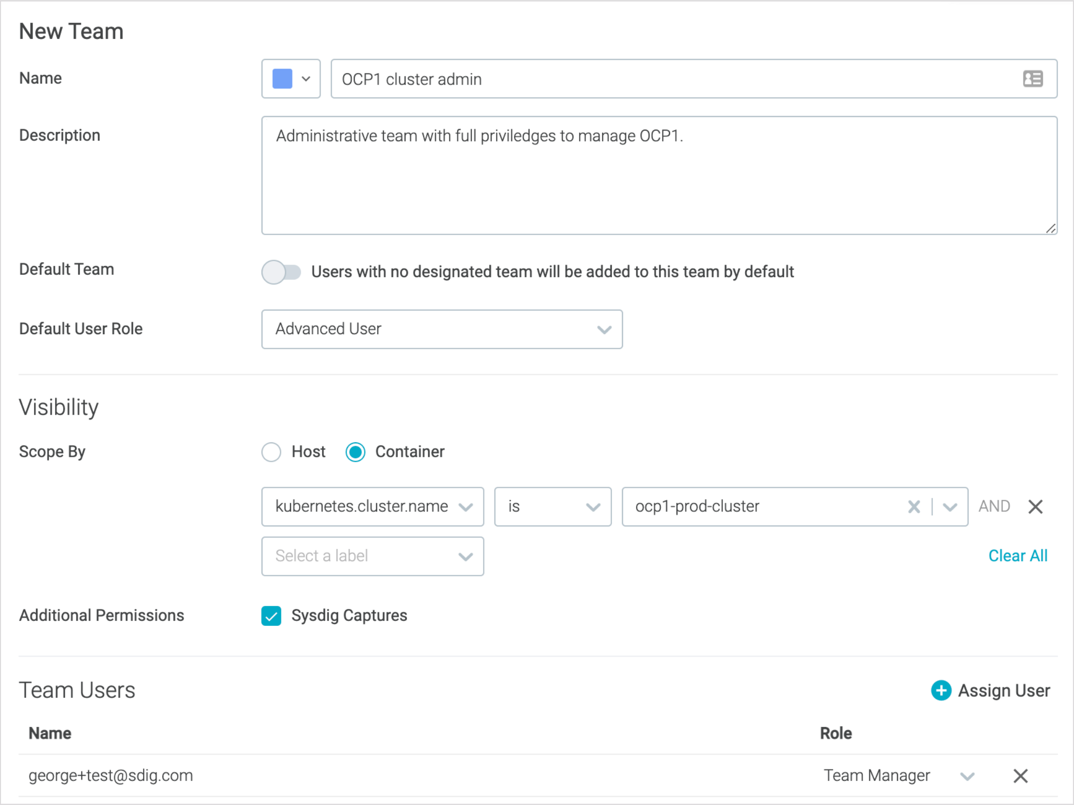
OpenShift leverages the Kubernetes RBAC system to define what users and services can do (create, read, update, delete) and communicate across any resource within the cluster (nodes, projects, deployments, pods, etc.).

Sysdig adds…

With Sysdig, you can define who can access any of the visibility, metrics, notifications, and security policies for your OpenShift deployments. Sysdig Teams enables the concept of service and metadata-based access control to complement the existing OpenShift authentication mechanisms.

With Sysdig Teams, administrators can define groups of users that have access to a specific service or limited set of services deployed on OpenShift. For example, an application owner might only see vulnerability scan results of images in a specific namespace. Limiting the exposure with access controls and providing a default configuration for each specific team helps streamline security operations.

Sysdig supports role-based access controls (RBAC) to define user privileges and provides federated access control across different teams in an organization. In addition to the admin role, a variety of access roles are available, including View Only, Standard User, Advanced User, and Team Manager.



New Team

Name: OCP1 cluster admin

Description: Administrative team with full priviledges to manage OCP1.

Default Team: Users with no designated team will be added to this team by default

Default User Role: Advanced User

Visibility

Scope By: ○ Host ● Container

kubernetes.cluster.name | is | ocp1-prod-cluster | AND

Select a label | Clear All

Additional Permissions: ☑ Sysdig Captures

Team Users | Assign User

Name	Role
george+test@sdig.com	Team Manager

Image Scanning

Container applications and infrastructure components are built on top of readily available packages, many of which are open-source software that might contain old library versions. It's important to know where these packages originally came from, who built them, and whether there are any known vulnerabilities inside them.

Red Hat provides...

Clair is the open-source engine that powers the Red Hat Quay container registry security scanner to detect vulnerabilities in OS packages across images within Red Hat Quay, and then notify developers as those issues are discovered. Red Hat Quay and Clair have boosted confidence in using containers in production, and organizations now want to get deeper into vulnerability scanning policies, security best practices, and regulatory compliance.

Sysdig adds...

Sysdig Secure embeds security and compliance across all stages of the Kubernetes lifecycle. Leveraging 15+ CVE threat feeds, Sysdig Secure provides a single workflow to detect vulnerabilities and security or compliance-related misconfigurations. As your teams build applications, Sysdig prevents vulnerable images from being pushed through your CI/CD pipeline and identifies new vulnerabilities in production.

Sysdig Secure provides additional scanning capabilities that extend beyond Quay/Clair default image scanning. When configured with your container registry, Sysdig Secure pulls images stored within the registry into the engine for analysis. Teams can scan for vulnerabilities, compliance checks, and misconfigurations before deployment. Vulnerabilities can be detected in base images, OS packages, and third-party libraries like Python packages from PIP, or Java JAR files that developers might be pulling into their application images before they hit production.

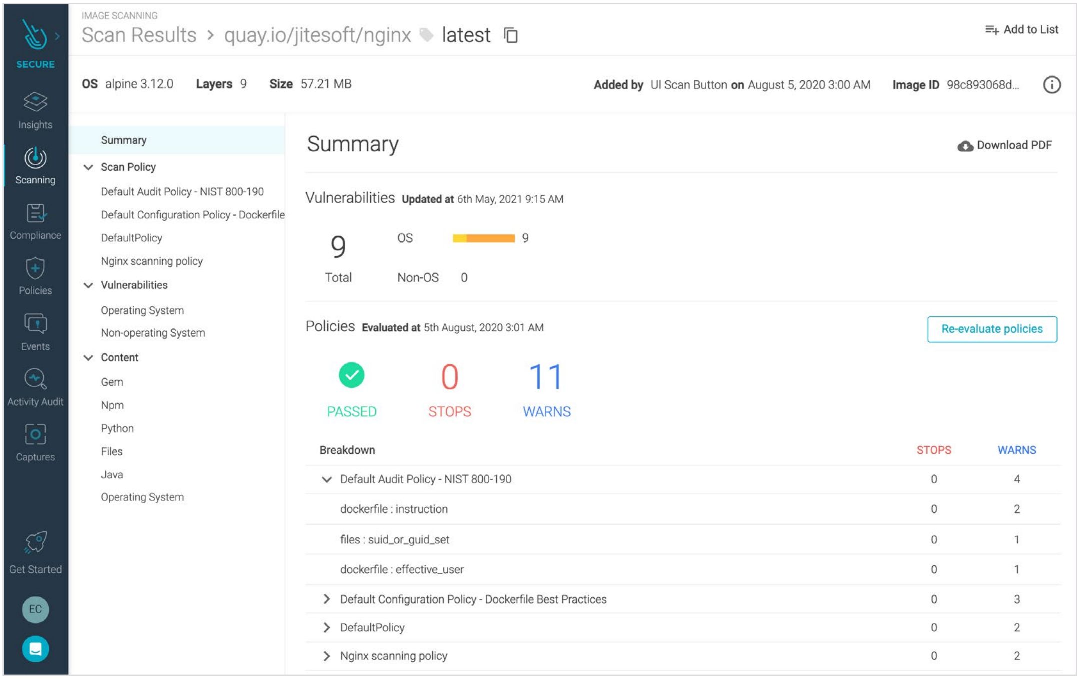
When it comes to pre-deployment scanning, Sysdig provides two container image scanning options.

- A standard approach that requires you to send your images to Sysdig for scanning. Post-scan, you can view the results within the Sysdig Secure UI.
- Local scanning, also known as inline scanning, scans images directly within your CI/CD pipeline, like Tekton or Red Hat OpenShift pipelines, or registry. This option enables a more secure approach as you don't need to share registry credentials or image contents outside of your environment. You also get scan results quickly by having the scan automated and reports generated directly within your pipeline and repository tools.

Sysdig Secure provides visibility into:

- Official OS package vulnerabilities.
- Unofficial package vulnerabilities.
- Configuration checks (e.g., exposing SSH in a Dockerfile, users running as root).
- Vulnerabilities in third-party libraries such as Javascript NPM modules, Python PiP, Ruby.
- GEM and Java JAR archives.
- Secrets, credentials like tokens, certificates, and other sensitive data.
- Known vulnerabilities and available updates.
- Metadata (e.g., size of an image).
- Compliance checks for frameworks like NIST 800-190, PCI, etc.

These artifacts are stored and evaluated against custom scanning policies that can be specified to a particular registry, repository, or image tag. Sysdig Secure scanning policies help detect vulnerabilities, misconfiguration, or compliance issues within your images and generate pass/fail results directly in the UI.



Scan Results > quay.io/jitesoft/nginx latest

OS alpine 3.12.0 **Layers** 9 **Size** 57.21 MB **Added by** UI Scan Button **on** August 5, 2020 3:00 AM **Image ID** 98c893068d...

Summary
Scan Policy
Default Audit Policy - NIST 800-190
Default Configuration Policy - Dockerfile
DefaultPolicy
Nginx scanning policy
Vulnerabilities
Operating System
Non-operating System
Content
Gem
Npm
Python
Files
Java
Operating System

Summary

Vulnerabilities **Updated at** 6th May, 2021 9:15 AM

9 Total — OS 9 — Non-OS 0

Policies **Evaluated at** 5th August, 2020 3:01 AM

PASSED — 0 STOPS — 11 WARNS

Breakdown	STOPS	WARNS
Default Audit Policy - NIST 800-190	0	4
dockerfile : instruction	0	2
files : suid_or_guid_set	0	1
dockerfile : effective_user	0	1
Default Configuration Policy - Dockerfile Best Practices	0	3
DefaultPolicy	0	2
Nginx scanning policy	0	2

Combining all of these capabilities, users can build policies like “Detect if any running image has a vulnerability classified as medium or high and there has been a fix available for more than seven days.”

DevOps and security teams can easily query across a catalog of images, packages, and CVEs, as well as check for advanced conditions like CVE age, fix available, software version, and more. Finally, these reports can be downloaded and shared (PDF/CSV) with vulnerability management teams, CISO’s, etc.

CI/CD pipeline security

CI/CD pipelines automate steps in your software delivery process, such as build and test, to help your teams deliver updates to your customers faster and more frequently. Embedding security into your delivery pipeline as you build applications helps you identify and address vulnerabilities faster, and keeps your developers productive.

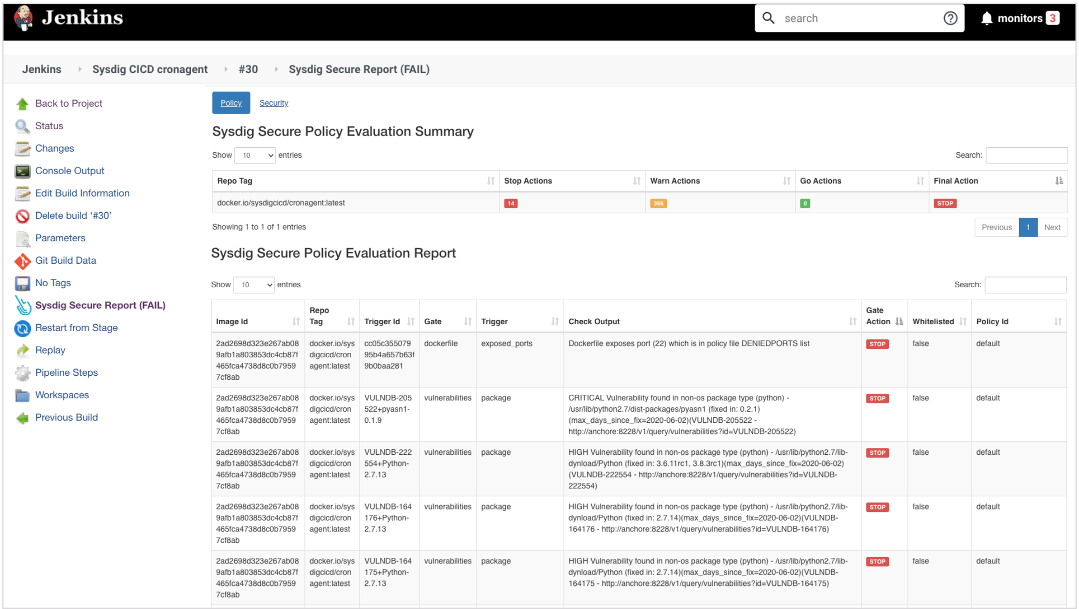
Red Hat provides...

OpenShift tightly integrates with Jenkins and Tekton (used in OpenShift 4.1+ Pipelines) to implement Continuous Integration/Continuous Delivery (CI/CD) pipelines. This allows developers to automate builds, code inspection, scanning, and test validation.

Sysdig adds...

Sysdig Secure image scanning integrates directly into your CI/CD pipeline of choice, including Jenkins, Tekton, Bamboo, GitLab, CircleCI, and more. You can catch vulnerabilities and misconfigurations in third-party libraries, official/unofficial OS packages, configuration checks, credential exposures, and metadata. Using Sysdig's inline scanning, you can detect issues before the images are even pushed to the registry.

Sysdig's scanning integration with CI/CD pipelines gives developers the information they need directly within their CI/CD tool to understand why a scan failed and what needs to be fixed. For non-critical policy violations, warnings will suggest what needs to be changed to improve the security of the container image without aborting the pipeline.



Sysdig Secure Policy Evaluation Summary

Show 10 entries Search:

Repo Tag	Stop Actions	Warn Actions	Go Actions	Final Action
docker.io/sysdigcicd/cronagent:latest	14	366	0	STOP

Showing 1 to 1 of 1 entries

Previous 1 Next

Sysdig Secure Policy Evaluation Report

Show 10 entries Search:

Image Id	Repo Tag	Trigger Id	Gate	Trigger	Check Output	Gate Action	Whitelisted	Policy Id
2ad2698d323e267ab089afb1a803853dc4cb87f465fca4738d8c0b79597cf8ab	docker.io/sysdigcicd/cronagent:latest	cc05c35507995b4a657b63f9b0baa281	dockerfile	exposed_ports	Dockerfile exposes port (22) which is in policy file DENIEDPORTS list	STOP	false	default
2ad2698d323e267ab089afb1a803853dc4cb87f465fca4738d8c0b79597cf8ab	docker.io/sysdigcicd/cronagent:latest	VULNDB-205522+pyasn1-0.1.9	vulnerabilities	package	CRITICAL Vulnerability found in non-os package type (python) - /usr/lib/python2.7/dist-packages/pyasn1 (fixed in: 0.2.1) (max_days_since_fix=2020-06-02)(VULNDB-205522 - http://anchore:8228/v1/query/vulnerabilities?id=VULNDB-205522)	STOP	false	default
2ad2698d323e267ab089afb1a803853dc4cb87f465fca4738d8c0b79597cf8ab	docker.io/sysdigcicd/cronagent:latest	VULNDB-222554+Python-2.7.13	vulnerabilities	package	HIGH Vulnerability found in non-os package type (python) - /usr/lib/python2.7/lib-dynload/Python (fixed in: 3.6.11rc1, 3.8.3rc1)(max_days_since_fix=2020-06-02) (VULNDB-222554 - http://anchore:8228/v1/query/vulnerabilities?id=VULNDB-222554)	STOP	false	default
2ad2698d323e267ab089afb1a803853dc4cb87f465fca4738d8c0b79597cf8ab	docker.io/sysdigcicd/cronagent:latest	VULNDB-164176+Python-2.7.13	vulnerabilities	package	HIGH Vulnerability found in non-os package type (python) - /usr/lib/python2.7/lib-dynload/Python (fixed in: 2.7.14)(max_days_since_fix=2020-06-02)(VULNDB-164176 - http://anchore:8228/v1/query/vulnerabilities?id=VULNDB-164176)	STOP	false	default
2ad2698d323e267ab089afb1a803853dc4cb87f465fca4738d8c0b79597cf8ab	docker.io/sysdigcicd/cronagent:latest	VULNDB-164175+Python-2.7.13	vulnerabilities	package	HIGH Vulnerability found in non-os package type (python) - /usr/lib/python2.7/lib-dynload/Python (fixed in: 2.7.14)(max_days_since_fix=2020-06-02)(VULNDB-164175 - http://anchore:8228/v1/query/vulnerabilities?id=VULNDB-164175)	STOP	false	default

Image assurance

Image assurance focuses on preventing unapproved images from being deployed in your container environment. This helps you reduce issues and errors by evaluating and verifying images based on your defined policies prior to running in production.

Red Hat provides...

OpenShift image policies and Kubernetes admission controllers can be used to prevent unapproved images from being deployed in your orchestrated container cluster. Requests to the Kubernetes API to run an image are evaluated against policies and images that fail to meet defined security requirements will not be allowed to start.

Sysdig adds...

OpenShift can check against Sysdig Secure to determine if an image is compliant with configured security policies before allowing or disallowing an image to run. When using the admission controller, this security validation decision is propagated back to the Kube API, which will reply to the original requester and only persist the object in the etcd database if the image passed the checks.

Registry Security

In addition to securing your container images, the security of your registry itself is another key step to reducing risk for your organization. Using RBAC to manage who can pull and push container images, as well as using a private registry, are some of the steps you can take to protect your organization.

Red Hat provides...

Red Hat has enabled a repository of certified containers for Red Hat products and partner offerings via the Red Hat Ecosystem Catalog, a public container registry hosted by Red Hat. Container content is monitored for vulnerabilities by Red Hat and updated regularly. At the same time, quay.io offers hosted public and private repositories for application container images in the Cloud.

If you are hosting images locally, OpenShift comes with its own registry. For more advanced requirements, Red Hat also provides Quay as a secure private container registry tailored for the enterprise.

Sysdig adds...

Regardless of the secure registry approach you select, Sysdig Secure container image scanning will help you ensure images are scanned for vulnerabilities. It supports all Docker v2 compatible registries, including CoreOS Quay, Amazon ECR, DockerHub Private Registries, Google Container Registry, Artifact Registry, JFrog Artifactory, Microsoft ACR, SuSE Portus, and VMware Harbor.

Compliance

Enterprise computing environments running microservices on OpenShift consist of hundreds or thousands of interconnected applications and services, as well as a large and diverse set of users. To maintain control over the security of this vast environment, a standard way to scan systems for compliance with security policies is needed.

Red Hat provides...

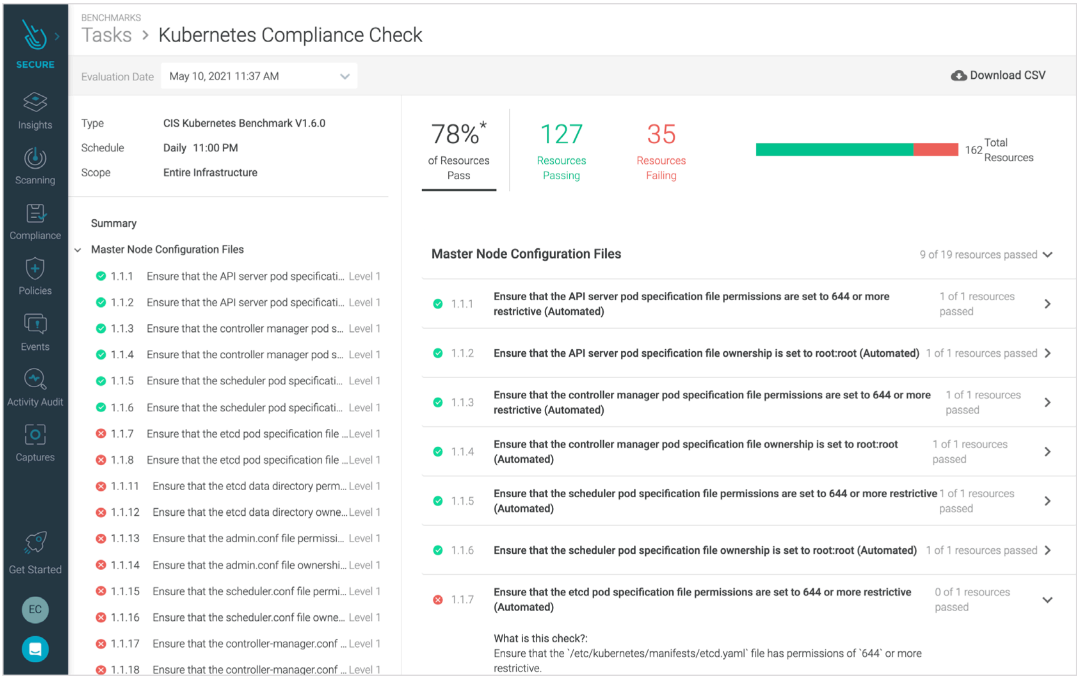
Red Hat Enterprise Linux and Red Hat CloudForms provide tools that allow for fully automated compliance audits. These tools, under the OpenSCAP project umbrella, are based on the Security Content Automation Protocol (SCAP) standard. Within these tools, you will find specific container-oriented checks, like oscap-docker, that perform CVE scans of containers and check them against predefined policies. This is done to validate security compliance content as well as generate reports and guides based on these scans and evaluations.

Sysdig adds...

Sysdig extends compliance across the container lifecycle for standards like NIST, PCI, and SOC2, and helps you track progress using compliance dashboards. Being able to validate that a deployment is compliant with desired configurations is one of the first compliance steps. But compliance requirements don't end there. Compliance for containers introduces unique requirements and should be implemented at various points:

- Checking against cloud, container, and infrastructure security best practices using Center for Internet Security (CIS) benchmarks for Kubernetes, Docker, and AWS.
- During build, mapping container image scanning policies to standards like NIST 800-190, NIST, 800-53, PCI, SOC2, and HIPAA.
- During runtime, using policies to continuously detect attack frameworks like MITRE ATT&CK or check compliance after deployment.
- Auditing any changes in your container environments, which is part of SOC2, PCI, ISO, and HIPAA requirements.

Sysdig helps you track progress using compliance dashboards and provides remediation guidance for correcting policy violations. This makes it faster to resolve configuration issues when they come up.



BENCHMARKS
Tasks > Kubernetes Compliance Check

Evaluation Date: May 10, 2021 11:37 AM

Type	CIS Kubernetes Benchmark V1.6.0
Schedule	Daily 11:00 PM
Scope	Entire Infrastructure

Summary

Master Node Configuration Files

✓	1.1.1	Ensure that the API server pod specificati...	Level 1
✓	1.1.2	Ensure that the API server pod specificati...	Level 1
✓	1.1.3	Ensure that the controller manager pod s...	Level 1
✓	1.1.4	Ensure that the controller manager pod s...	Level 1
✓	1.1.5	Ensure that the scheduler pod specificati...	Level 1
✓	1.1.6	Ensure that the scheduler pod specificati...	Level 1
☒	1.1.7	Ensure that the etcd pod specification file ...	Level 1
☒	1.1.8	Ensure that the etcd pod specification file ...	Level 1
☒	1.1.11	Ensure that the etcd data directory perm...	Level 1
☒	1.1.12	Ensure that the etcd data directory owne...	Level 1
☒	1.1.13	Ensure that the admin.conf file permissi...	Level 1
☒	1.1.14	Ensure that the admin.conf file ownershi...	Level 1
☒	1.1.15	Ensure that the scheduler.conf file permi...	Level 1
☒	1.1.16	Ensure that the scheduler.conf file owne...	Level 1
☒	1.1.17	Ensure that the controller-manager.conf ...	Level 1
☒	1.1.18	Ensure that the controller-manager.conf ...	Level 1

78%* of Resources Pass | 127 Resources Passing | 35 Resources Failing | 162 Total Resources

Master Node Configuration Files — 9 of 19 resources passed

✓	1.1.1	Ensure that the API server pod specification file permissions are set to 644 or more restrictive (Automated)	1 of 1 resources passed
✓	1.1.2	Ensure that the API server pod specification file ownership is set to root:root (Automated)	1 of 1 resources passed
✓	1.1.3	Ensure that the controller manager pod specification file permissions are set to 644 or more restrictive (Automated)	1 of 1 resources passed
✓	1.1.4	Ensure that the controller manager pod specification file ownership is set to root:root (Automated)	1 of 1 resources passed
✓	1.1.5	Ensure that the scheduler pod specification file permissions are set to 644 or more restrictive (Automated)	1 of 1 resources passed
✓	1.1.6	Ensure that the scheduler pod specification file ownership is set to root:root (Automated)	1 of 1 resources passed
☒	1.1.7	Ensure that the etcd pod specification file permissions are set to 644 or more restrictive (Automated)	0 of 1 resources passed

What is this check?:
Ensure that the `/etc/kubernetes/manifests/etcd.yaml` file has permissions of `644` or more restrictive.

Sysdig Secure gives you the tools to implement container image security and compliance best practices for NIST SP 800-190, PCI DSS, Dockerfile, and more. Using Sysdig Secure container image scanning policies, you can validate cloud compliance and enforce best practices at the image level, including:

- Limiting image size.
- Blacklisting GPlv2 licenses.
- Ensuring containers only use trusted base images and necessary packages.

Sysdig provides runtime compliance assurance by translating leading security standards like NIST SP 800-190, PCI DSS, CIS benchmarks, HIPAA, GDPR, or the MITRE ATT&CK framework into a set of up-to-date security policies. It will analyze container behavior after deployment, auditing any runtime drift. Sysdig taps into any executed command on the system (both at the host and inside any container, like docker exec or oc attach), or the OpenShift API for auditing purposes (audit secret resources access, requests by unauthorized users, etc.).

When a new high/critical CVE is published, you can assess your exposure immediately. Affected services and accountable teams can be quickly identified. Developers or application owners are identified using Kubernetes or cloud metadata, like service, deployment, or application, and alerted to view their images and vulnerabilities.

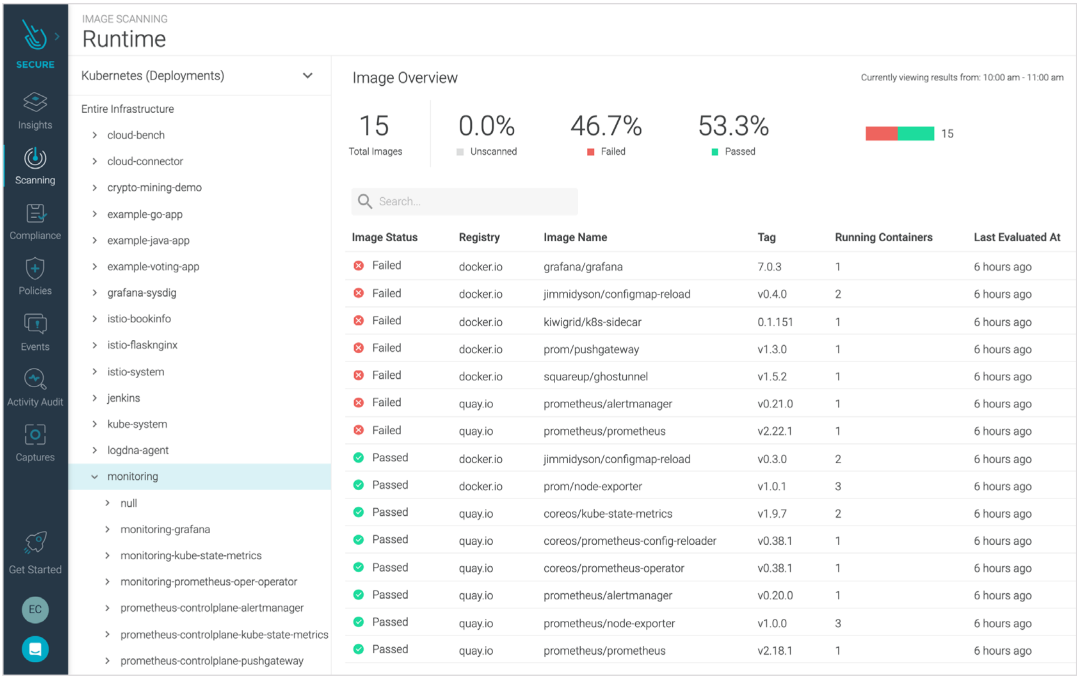


IMAGE SCANNING

Runtime

Image Overview

Currently viewing results from: 10:00 am - 11:00 am

15 Total Images

0.0% Unscanned

46.7% Failed

53.3% Passed

15

Image Status	Registry	Image Name	Tag	Running Containers	Last Evaluated At
Failed	docker.io	grafana/grafana	7.0.3	1	6 hours ago
Failed	docker.io	jimmidyson/configmap-reload	v0.4.0	2	6 hours ago
Failed	docker.io	kiwigrid/k8s-sidecar	0.1.151	1	6 hours ago
Failed	docker.io	prom/pushgateway	v1.3.0	1	6 hours ago
Failed	docker.io	squareup/ghostunnel	v1.5.2	1	6 hours ago
Failed	quay.io	prometheus/alertmanager	v0.21.0	1	6 hours ago
Failed	quay.io	prometheus/prometheus	v2.22.1	1	6 hours ago
Passed	docker.io	jimmidyson/configmap-reload	v0.3.0	2	6 hours ago
Passed	docker.io	prom/node-exporter	v1.0.1	3	6 hours ago
Passed	quay.io	coreos/kube-state-metrics	v1.9.7	2	6 hours ago
Passed	quay.io	coreos/prometheus-config-reloader	v0.38.1	1	6 hours ago
Passed	quay.io	coreos/prometheus-operator	v0.38.1	1	6 hours ago
Passed	quay.io	prometheus/alertmanager	v0.20.0	1	6 hours ago
Passed	quay.io	prometheus/node-exporter	v1.0.0	3	6 hours ago
Passed	quay.io	prometheus/prometheus	v2.18.1	1	6 hours ago

Network security

The shift of applications to containers and the cloud is a catalyst for rethinking your security model. Many cloud teams are taking a Zero Trust approach, requiring authentication and authorization even for networks internal to their organization. The ability to segment, isolate, and control networks is a critical point of control for Zero Trust and is increasingly essential to achieving more effective security in container and Kubernetes environments.

Without the right tools, DevOps teams will struggle to see how their containerized apps are communicating, and may miss malicious attempts that take advantage of open network policies. Applying a Zero Trust network security model in Kubernetes is challenging without knowing how applications are being used.

Red Hat provides...

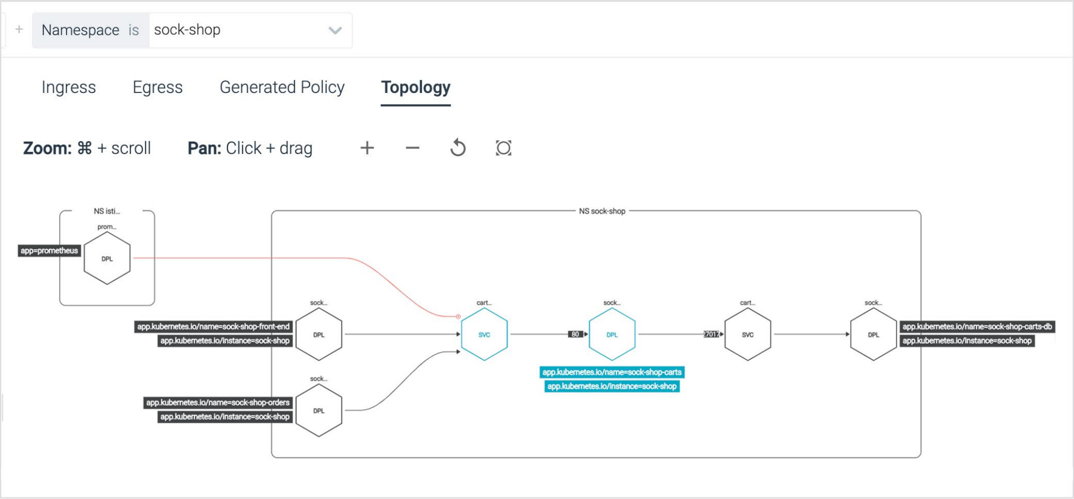
OpenShift uses software-defined networking (SDN) to provide cluster networking that enables communication between containers across the cluster. Network security can be managed at several levels. At the pod level, network namespaces can prevent containers from seeing other pods or the host system by restricting network access.

Kubernetes network policies in OpenShift give you control over allowing and rejecting connections. You can manage ingress and egress traffic to and from your containerized applications. Network policy mode, by default, makes all pods in a project accessible from other pods and network endpoints. To isolate one or more pods in a project, you can create NetworkPolicy objects in that project to indicate the allowed incoming connections. Using multi-tenant mode, you can provide project-level isolation for pods and services.

Sysdig adds...

Kubernetes network policies provide a native option for controlling network traffic within your clusters to achieve network security. With native controls, you get better performance, reliability, and security because Kubernetes enforces network microsegmentation. The challenge, however, is that Kubernetes network policies can be hard to implement without the right application knowledge and Kubernetes expertise. Sysdig helps remove these barriers to simplify implementing Zero Trust network security with Kubernetes controls.

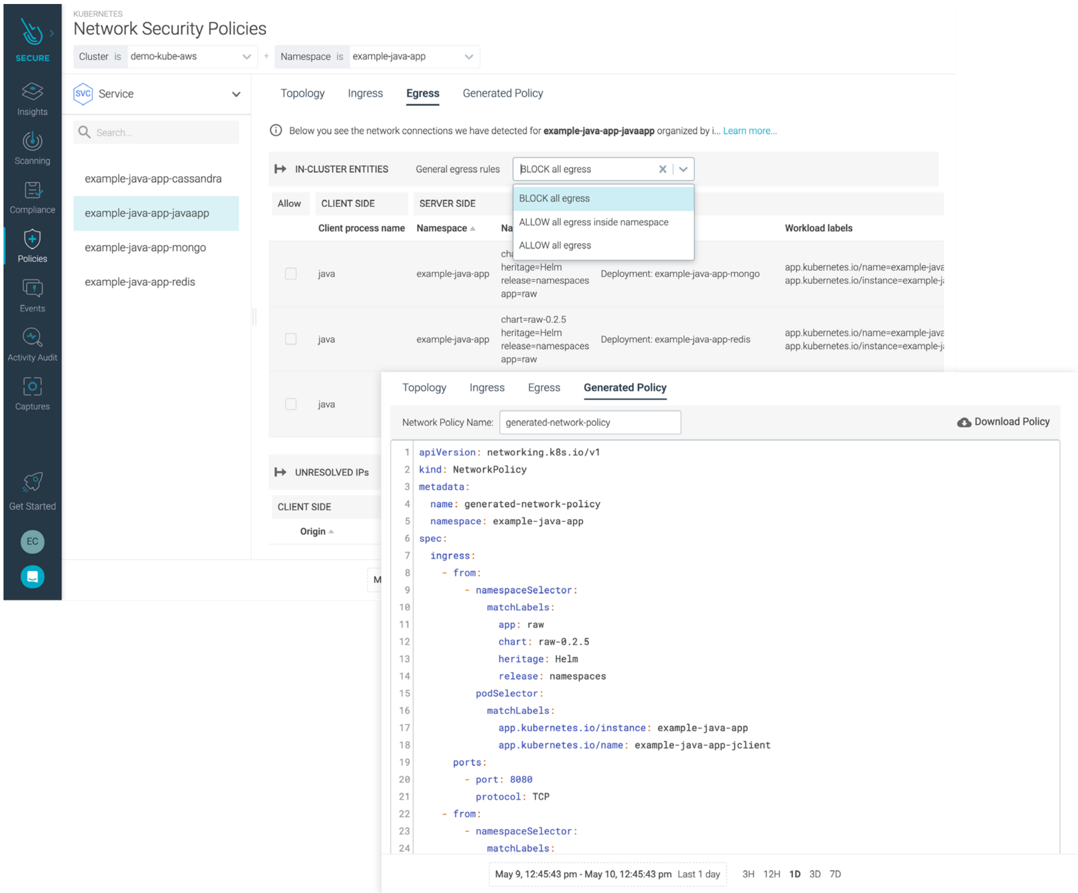
Sysdig Secure automatically discovers all network traffic for OpenShift pods, services, and applications through visibility into system calls. The data is auto-tagged with Kubernetes context and labels, which are used to simplify your experience when implementing Kubernetes network policies with OpenShift.



Namespace is sock-shop

+
Ingress
Egress
Generated Policy
Topology
Zoom: ⌘ + scroll
Pan: Click + drag
NS isti...
prom...
app=prometheus
DPL
NS sock-shop
sock...
app.kubernetes.io/name=sock-shop-front-end
app.kubernetes.io/instance=sock-shop
DPL
sock...
app.kubernetes.io/name=sock-shop-orders
app.kubernetes.io/instance=sock-shop
DPL
cart...
SVC
80
sock...
DPL
app.kubernetes.io/name=sock-shop-carts
app.kubernetes.io/instance=sock-shop
27017
cart...
SVC
sock...
DPL
app.kubernetes.io/name=sock-shop-carts-db
app.kubernetes.io/instance=sock-shop

Dynamic topology maps let you visualize all network communication between apps and services, and drill down into the traffic flow over a particular time frame. Using this information in a simple UI, you can apply segmentation and refine network policies to allow or block connections. Sysdig will automatically generate a YAML file that you can use to apply the policy to your OpenShift cluster.



KUBERNETES

Network Security Policies

Cluster is demo-kube-aws + Namespace is example-java-app

Service

Search...

- example-java-app-cassandra
- example-java-app-javaapp
- example-java-app-mongo
- example-java-app-redis

Topology | Ingress | **Egress** | Generated Policy

Below you see the network connections we have detected for **example-java-app-javaapp** organized by i... Learn more...

IN-CLUSTER ENTITIES General egress rules: BLOCK all egress

- BLOCK all egress
- ALLOW all egress inside namespace
- ALLOW all egress

Allow	CLIENT SIDE	SERVER SIDE			
	Client process name	Namespace	Na		Workload labels
□	java	example-java-app	cha heritage=Helm release=namespaces app=raw	Deployment: example-java-app-mongo	app.kubernetes.io/name=example-java app.kubernetes.io/instance=example-j
□	java	example-java-app	chart=raw-0.2.5 heritage=Helm release=namespaces app=raw	Deployment: example-java-app-redis	app.kubernetes.io/name=example-java app.kubernetes.io/instance=example-j
□	java				

UNRESOLVED IPs

CLIENT SIDE

Origin

Topology | Ingress | Egress | **Generated Policy**

Network Policy Name: generated-network-policy Download Policy

```
apiVersion: networking.k8s.io/v1
kind: NetworkPolicy
metadata:
  name: generated-network-policy
  namespace: example-java-app
spec:
  ingress:
    - from:
        - namespaceSelector:
            matchLabels:
              app: raw
              chart: raw-0.2.5
              heritage: Helm
              release: namespaces
          podSelector:
            matchLabels:
              app.kubernetes.io/instance: example-java-app
              app.kubernetes.io/name: example-java-app-jclient
      ports:
        - port: 8080
          protocol: TCP
    - from:
        - namespaceSelector:
            matchLabels:
```

May 9, 12:45:43 pm - May 10, 12:45:43 pm Last 1 day 3H 12H **1D** 3D 7D

In addition, Sysdig Secure can fingerprint every connection – and the processes that are establishing connections. This Audit Tap capability helps cloud teams investigate network activity at a fine-grained level with full visibility into context, including labels. If your organization is subject to regulations, such as NIST and PCI, you can use this capability along with network segmentation to meet compliance requirements.

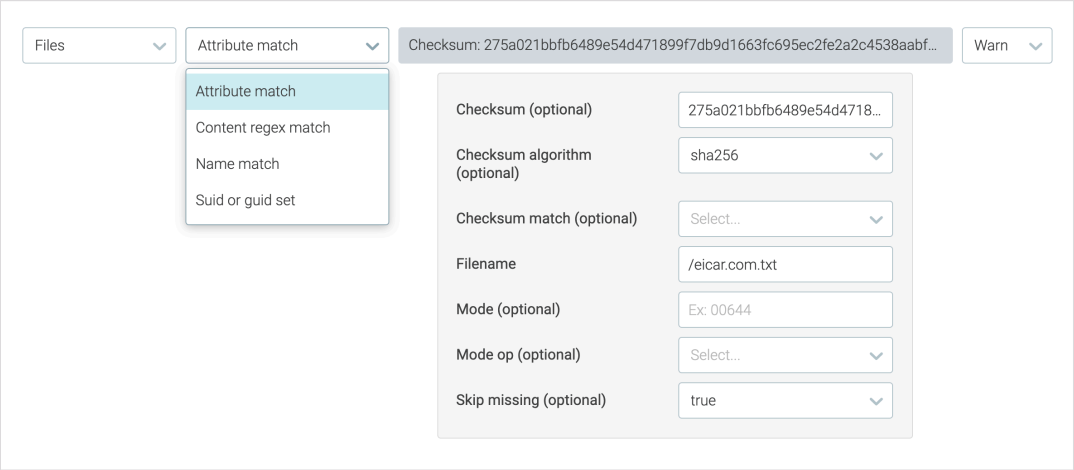
Using Sysdig to enable Zero Trust network security based on an open, standards-based approach vetted by the community delivers better performance, reliability, and security because Kubernetes provides enforcement. This eliminates the need for man-in-the- middle enforcement mechanisms. By providing an easy-to-use interface and automating guardrails for teams who may lack Kubernetes expertise, Sysdig helps OpenShift users save time and reduce network security risk.

File Integrity Monitoring

File integrity monitoring (FIM) gives you visibility into all of your sensitive file-related activity. It's used to detect tampering of critical system files, directories, and unauthorized changes, regardless of whether the activity is a malicious attack or an unplanned operational activity.

With Sysdig Secure, you can scan for specific file attributes as part of your image scanning policy within your CI/CD pipelines. This allows you to fail builds early if FIM policies aren't met. The file integrity monitoring policy allows you to:

- Check if a file exists or is missing, and trigger alerts based on the condition.
- Validate a specific file against its SHA256 hash. Any modification to binaries in your containers is flagged as suspicious and potentially dangerous.
- Validate file permissions. For example, you can be alerted if a file has an executable bit where it's unexpected.
- Check for file names based on regex.
- Inspect contents, looking for exposed passwords and credential leaks.



Files | Attribute match | Checksum: 275a021bbfb6489e54d471899f7db9d1663fc695ec2fe2a2c4538aabf... | Warn

Checksum (optional)	275a021bbfb6489e54d4718...
Checksum algorithm (optional)	sha256
Checksum match (optional)	Select...
Filename	/eicar.com.txt
Mode (optional)	Ex: 00644
Mode op (optional)	Select...
Skip missing (optional)	true

You can also implement FIM policies at runtime that alert on any suspicious changes to a filesystem. These are common file integrity monitoring checks that you should include as rules to enforce a strong security posture:

- Creation or removal of files or directories.
- Renaming of files or directories.
- Changes to file or directory security settings such as permissions, ownership, and inheritance.
- Changes to the files of a container.
- Modification of files below the container's path.
- Deletion of bash history.

Beyond generating robust reports, the Sysdig platform translates security benchmarks into a set of security metrics and dashboards. Internal and external compliance and audit teams can analyze their security posture, quickly visualize patterns and trends, and gain valuable insights into their compliance posture to:

- Compare your security posture to any previous point in time.
- Understand the risk and compliance posture across applications and environments.
- Alert when a compliance check falls below the accepted policy.
- Detect any configuration drift across your OpenShift cluster and containers.

Runtime Security

Scanning your containers once during the CI/CD process or from your OpenShift registry is not enough. While known software vulnerabilities are detected, several security threats, by their very nature, only manifest during runtime, including:

- Zero-day vulnerabilities and non-public vulnerabilities specific to your own software.
- Software bugs causing erratic behavior or resource leaking.
- Internal privilege escalation attempts or hidden/embedded malware.

To protect against these security threats, there are a number of runtime security techniques you should implement for your OpenShift clusters and cloud environment.

Security monitoring

Gaining visibility across both monitoring and security for OpenShift container services is necessary for a successful transformation journey. For example, the security team needs to know if a crypto-mining or denial of service (DoS) attack can be further explained by an abnormal deviation in a particular performance metric.

Additionally, once in production, it's important to reduce risk by configuring applications with the minimum privilege and access permissions. At the same time, you need to be able to create and maintain a runtime policy that observes workload behavior and looks for anomalous activity, blocking any threats and attacks not caught in your CI/CD or registry scanning.

Threat detection

Detecting threats to your running containers requires visibility into what's happening inside containers and across microservices. The challenge, as noted previously, is the black-box nature of containers can impede visibility, making it harder to identify anomalies and unexpected behavior. With the right tools, however, you can overcome this limitation and successfully identify and block security threats.

OpenShift adds...

OpenShift users can use Falco, the open source CNCF® runtime security project, to enable detection rules to identify unexpected and anomalous activity at runtime. In addition, teams managing multiple clusters using Red Hat Advanced Cluster Management can use a custom policy to confirm Falco is deployed on all targeted clusters, ensuring runtime security visibility is active and available.

In addition to default rules available with Falco, community-sourced and curated rules are available on the Cloud Native Security Hub.

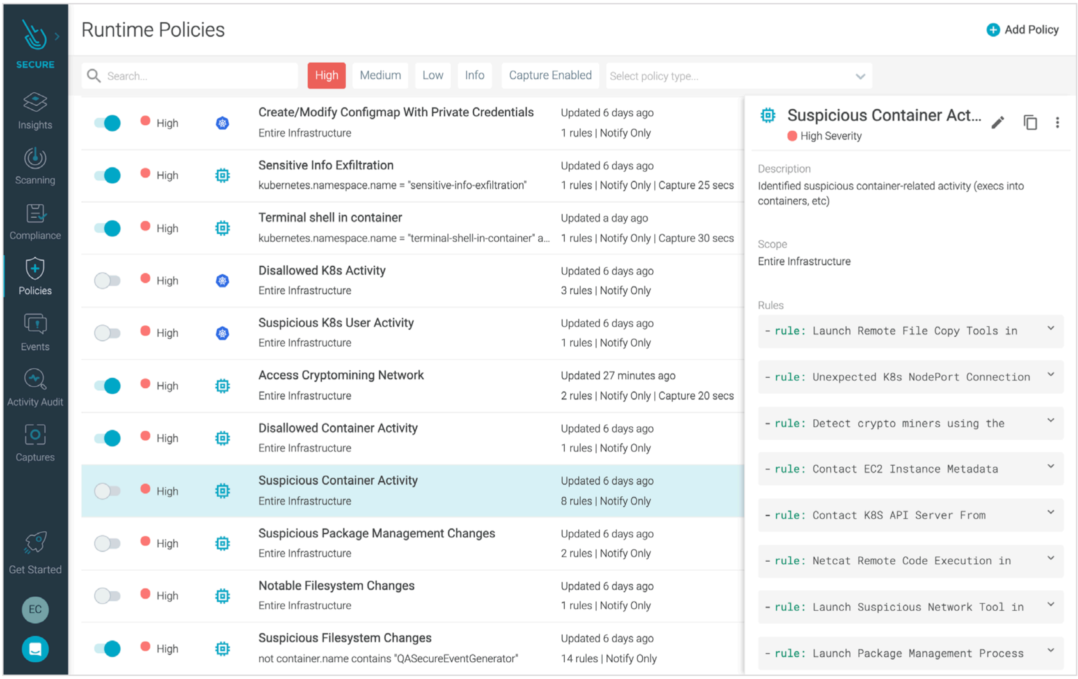
Sysdig adds...

Sysdig Secure leverages the Falco engine under the hood for runtime security and cloud threat detection. Sysdig Secure saves time in creating and maintaining policies using a Web UI for easier policy creation and customization at scale.

Default policies are available out-of-the-box along with more than 200 rules that simplify the job of customizing security to meet your requirements. Using Sysdig Secure policies, you can implement runtime security quickly and easily and detect threats to your OpenShift and cloud environment in minutes.

Sysdig Secure runtime policies include:

- Container runtime security policies for regulatory container compliance standards including NIST, PCI, SOC2, HIPAA, CIS, and the MITRE ATT&CK framework.
- Runtime detection of the most pervasive container attacks: cryptomining, secrets exfiltration, container isolation breaches, and lateral movements.
- Security monitoring for unexpected process activity, outbound connections, and terminal shell sessions.
- Cloud log detection rules to identify suspicious cloud activity when running OpenShift on a public cloud.



Runtime Policies

Add Policy

Search... High Medium Low Info Capture Enabled Select policy type...

High	Create/Modify Configmap With Private Credentials Entire Infrastructure	Updated 6 days ago 1 rules | Notify Only
High	Sensitive Info Exfiltration kubernetes.namespace.name = "sensitive-info-exfiltration"	Updated 6 days ago 1 rules | Notify Only | Capture 25 secs
High	Terminal shell in container kubernetes.namespace.name = "terminal-shell-in-container" a...	Updated a day ago 1 rules | Notify Only | Capture 30 secs
High	Disallowed K8s Activity Entire Infrastructure	Updated 6 days ago 3 rules | Notify Only
High	Suspicious K8s User Activity Entire Infrastructure	Updated 6 days ago 1 rules | Notify Only
High	Access Cryptomining Network Entire Infrastructure	Updated 27 minutes ago 2 rules | Notify Only | Capture 20 secs
High	Disallowed Container Activity Entire Infrastructure	Updated 6 days ago 1 rules | Notify Only
High	Suspicious Container Activity Entire Infrastructure	Updated 6 days ago 8 rules | Notify Only
High	Suspicious Package Management Changes Entire Infrastructure	Updated 6 days ago 2 rules | Notify Only
High	Notable Filesystem Changes Entire Infrastructure	Updated 6 days ago 1 rules | Notify Only
High	Suspicious Filesystem Changes not container.name contains "QASecureEventGenerator"	Updated 6 days ago 14 rules | Notify Only

Suspicious Container Act...
High Severity

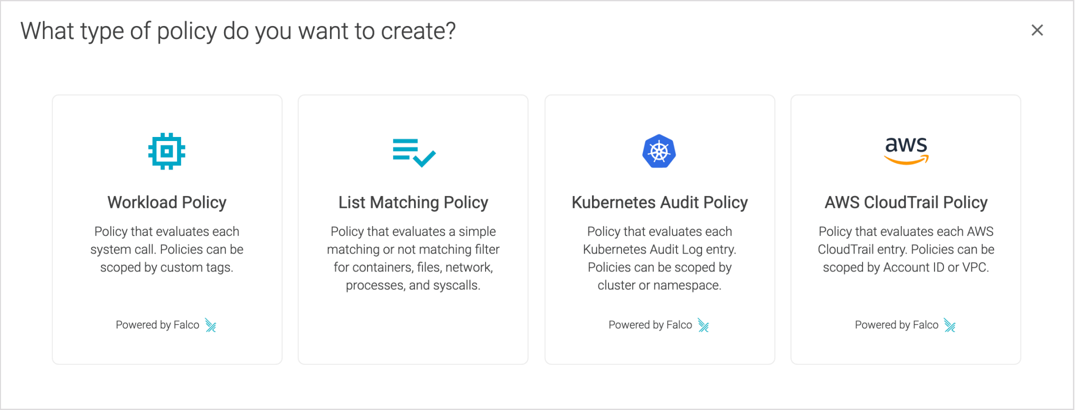
Description
Identified suspicious container-related activity (execs into containers, etc)

Scope
Entire Infrastructure

Rules

```
- rule: Launch Remote File Copy Tools in
- rule: Unexpected K8s NodePort Connection
- rule: Detect crypto miners using the
- rule: Contact EC2 Instance Metadata
- rule: Contact K8S API Server From
- rule: Netcat Remote Code Execution in
- rule: Launch Suspicious Network Tool in
- rule: Launch Package Management Process
```

With an extensible policy engine, platform operations and security teams can create and customize their own rules through a visual interface to build fine-tuned policies to match unique requirements.



What type of policy do you want to create?

Workload Policy
Policy that evaluates each system call. Policies can be scoped by custom tags.
Powered by Falco

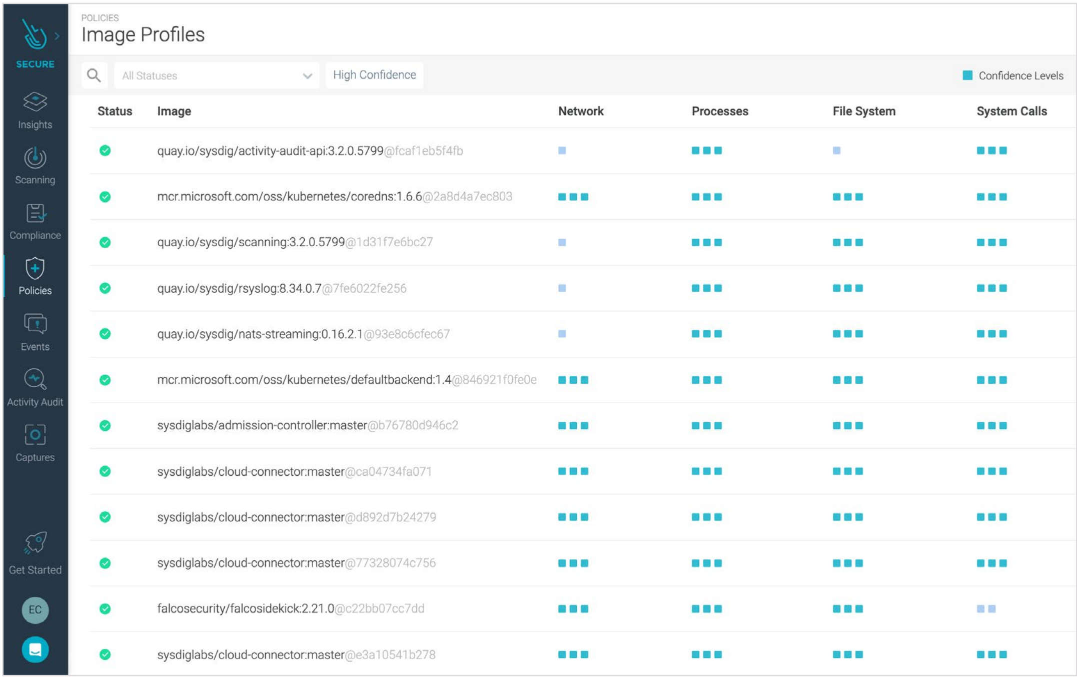
List Matching Policy
Policy that evaluates a simple matching or not matching filter for containers, files, network, processes, and syscalls.

Kubernetes Audit Policy
Policy that evaluates each Kubernetes Audit Log entry. Policies can be scoped by cluster or namespace.
Powered by Falco

AWS CloudTrail Policy
Policy that evaluates each AWS CloudTrail entry. Policies can be scoped by Account ID or VPC.
Powered by Falco

Runtime image profiling

To ease the burden of creating and maintaining runtime security in large-scale environments, Sysdig Secure provides runtime image profiling. Image profiling automatically models, analyzes, and learns container runtime behavior to create a comprehensive container runtime profile and automatically builds policies for you. This includes analyzing kubeapiserver activity and syscalls while enriching them with various metadata, including Kubernetes and cloud labels. This approach enhances anomaly detection through machine learning and helps you block threats before they propagate.

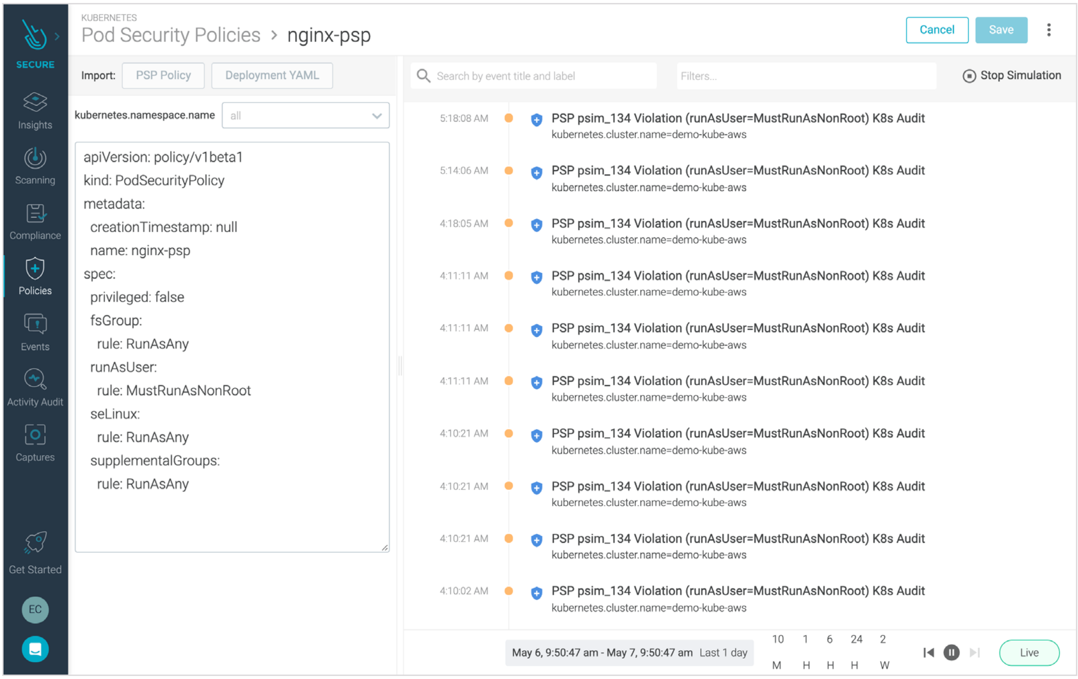


Status	Image	Network	Processes	File System	System Calls
✓	quay.io/sysdig/activity-audit-api:3.2.0.5799@fcaf1eb5f4fb	■	■■■	■	■■■
✓	mcr.microsoft.com/oss/kubernetes/coredns:1.6.6@2a8d4a7ec803	■■■	■■■	■■■	■■■
✓	quay.io/sysdig/scanning:3.2.0.5799@1d31f7e6bc27	■	■■■	■■■	■■■
✓	quay.io/sysdig/rsyslog:8.34.0.7@7fe6022fe256	■	■■■	■■■	■■■
✓	quay.io/sysdig/nats-streaming:0.16.2.1@93e8c6cfec67	■	■■■	■■■	■■■
✓	mcr.microsoft.com/oss/kubernetes/defaultbackend:1.4@846921f0fe0e	■■■	■■■	■■■	■■■
✓	sysdiglabs/admission-controller:master@b76780d946c2	■■■	■■■	■■■	■■■
✓	sysdiglabs/cloud-connector:master@ca04734fa071	■■■	■■■	■■■	■■■
✓	sysdiglabs/cloud-connector:master@d892d7b24279	■■■	■■■	■■■	■■■
✓	sysdiglabs/cloud-connector:master@77328074c756	■■■	■■■	■■■	■■■
✓	falcosecurity/falcosidekick:2.21.0@c22bb07cc7dd	■■■	■■■	■■■	■■
✓	sysdiglabs/cloud-connector:master@e3a10541b278	■■■	■■■	■■■	■■■

Threat prevention with Kubernetes native controls

Sysdig prevents threats using Kubernetes native controls, such as Pod Security Policies (PSPs). The Kubernetes Policy Advisor automates the generation of PSPs and validates them pre-deployment so they don't break applications when applied. This allows users to adopt PSPs in production environments quickly and easily. PSPs also provide a Kubernetes native control mechanism to prevent threats without impacting performance, unlike agents that have to intercept every action on the host.

Sysdig Secure leverages Kubernetes-native controls like PSP for enforcement. You can read more about it on the blog Pod Security Policies in production with Sysdig's Kubernetes Policy Advisor and learn about Sysdig runtime security capabilities here.



kubernetes.namespace.name: all

```
apiVersion: policy/v1beta1
kind: PodSecurityPolicy
metadata:
 creationTimestamp: null
 name: nginx-psp
spec:
 privileged: false
 fsGroup:
  rule: RunAsAny
 runAsUser:
  rule: MustRunAsNonRoot
 seLinux:
  rule: RunAsAny
 supplementalGroups:
  rule: RunAsAny
```

```
5:18:08 AM  PSP psim_134 Violation (runAsUser=MustRunAsNonRoot) K8s Audit
            kubernetes.cluster.name=demo-kube-aws
5:14:06 AM  PSP psim_134 Violation (runAsUser=MustRunAsNonRoot) K8s Audit
            kubernetes.cluster.name=demo-kube-aws
4:18:05 AM  PSP psim_134 Violation (runAsUser=MustRunAsNonRoot) K8s Audit
            kubernetes.cluster.name=demo-kube-aws
4:11:11 AM  PSP psim_134 Violation (runAsUser=MustRunAsNonRoot) K8s Audit
            kubernetes.cluster.name=demo-kube-aws
4:11:11 AM  PSP psim_134 Violation (runAsUser=MustRunAsNonRoot) K8s Audit
            kubernetes.cluster.name=demo-kube-aws
4:11:11 AM  PSP psim_134 Violation (runAsUser=MustRunAsNonRoot) K8s Audit
            kubernetes.cluster.name=demo-kube-aws
4:10:21 AM  PSP psim_134 Violation (runAsUser=MustRunAsNonRoot) K8s Audit
            kubernetes.cluster.name=demo-kube-aws
4:10:21 AM  PSP psim_134 Violation (runAsUser=MustRunAsNonRoot) K8s Audit
            kubernetes.cluster.name=demo-kube-aws
4:10:21 AM  PSP psim_134 Violation (runAsUser=MustRunAsNonRoot) K8s Audit
            kubernetes.cluster.name=demo-kube-aws
4:10:02 AM  PSP psim_134 Violation (runAsUser=MustRunAsNonRoot) K8s Audit
            kubernetes.cluster.name=demo-kube-aws
```

May 6, 9:50:47 am - May 7, 9:50:47 am Last 1 day

10 M 1 H 6 H 24 H 2 W

With Sysdig Secure, operations and security teams can ease the burden of creating container security policies, and gain more transparency and assurance since they have greater control of what's happening under the hood.

Cloud Security Posture Management

Threat research conducted by Sysdig shows that having a single view across cloud, workloads, and containers speeds the time to both detect and respond to lateral movement attacks, a common technique used in the majority of security breaches.

Using different cloud and container security tools complicates security operations as it requires manual correlation of different data sources to fully understand a breach and uncover the systems impacted. Sysdig pairs Cloud Security Posture Management (CSPM) and cloud threat detection with cloud workload protection, including container and Kubernetes security features in a single platform.

By unifying the incident timeline and adding risk-based insights, Sysdig helps customers who run OpenShift on a public cloud reduce the time to detect threats across cloud services and OpenShift containers from weeks to hours. Cloud teams can see exactly where the attacker started and each step they took as they moved through the environment.

Sysdig’s CSPM capabilities include:

- **Cloud Asset Discovery** – Key to cloud security is visibility into the assets in-use and interacting with your environment. Organizations need an inventory of assets to ensure proper configuration and security management.
- **Static Configuration Management -** Cloud service usage and features change continuously. Settings and configurations may need to be updated as cloud use evolves. Continuous cloud configuration monitoring and audit help detect security and compliance violations across cloud infrastructure and services to stay ahead of threats and vulnerabilities.
- **Threat Detection with cloud logs -** As your cloud infrastructure grows, staying on top of activity and changes is challenging. Cloud activity logging services, like AWS CloudTrail and Google Cloud logs, capture the information you need. However, as the amount of events and logs increase, performing manual analysis is time-consuming and difficult. Automating the evaluation of events in real time, using security rules to continuously detect and report changes and suspicious cloud activity, helps cloud teams address issues quickly and avoid potentially major consequences.

If you’re running OpenShift in a public cloud like AWS or Google, visit our Continuous Cloud Security Posture Management page to learn more about Sysdig’s Cloud Security Posture Management capabilities.

Monitoring Red Hat OpenShift

Containers are short-lived, dynamic, and churn constantly. Once a container dies, everything inside is gone. You cannot SSH or look at logs, and most of the traditional tools used for monolithic applications are of little help when something goes wrong. Containers are great for operations as you can package and isolate applications to consistently deploy them everywhere, but at the same time, this makes them black boxes that are hard to troubleshoot.

Monitoring the dynamic nature of container-based applications is critical for high availability and performance of cloud services. Microservice architectures running on containers make applications faster to develop and easier to scale, accelerating innovation and time-to-market for new features. As the number of microservices grows, it can become difficult to ensure visibility inside these environments. Microservices-based applications can be distributed across multiple instances, and deployed across multi-cloud infrastructure as needed. Monitoring the Kubernetes orchestration state as well as containers is key to understanding the health, performance, and state of your service instances.

Red Hat provides...

OpenShift includes a pre-configured, pre-installed, and self-updating monitoring stack based on Prometheus, that provides monitoring for core platform components. Prometheus is an open-source systems monitoring toolkit that enables you to gather metrics and data that you can use to track various aspects of your OpenShift environment. A set of alerts are included by default that notify administrators about issues with a cluster. Default dashboards in the OpenShift web console include visual representations of cluster metrics to help you to understand the state of your cluster.

With OpenShift Container Platform 4.7, cluster administrators can optionally enable monitoring for user-defined projects. With this feature, cluster administrators, developers, and other users can specify how services and pods are monitored in their own projects. You can then query metrics, review dashboards, and manage alerting rules for your own projects in the OpenShift Container Platform web console.

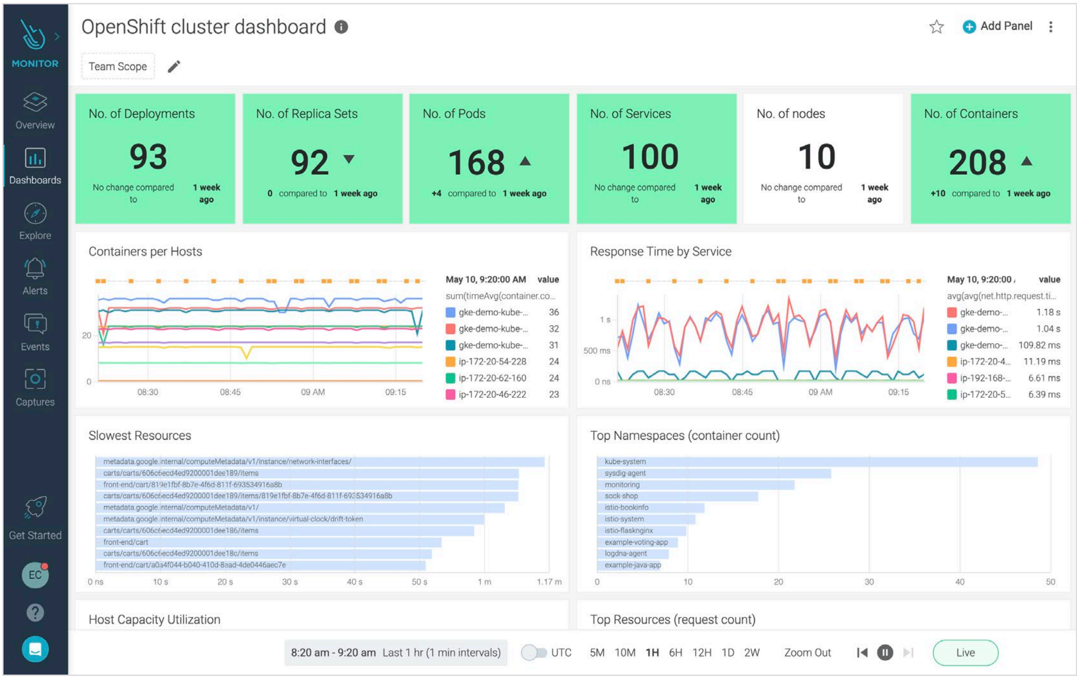
Sysdig Adds....

Sysdig Monitor allows you to maximize the performance and availability of your cloud infrastructure, services, and applications. It provides cloud and container monitoring at scale with full Prometheus compatibility, giving you deep visibility into rapidly changing cloud-native environments. You can resolve issues faster by using granular data derived from system calls, Prometheus, and other data sources that are enriched with cloud and Kubernetes context. This helps you eliminate silos by unifying data across teams for OpenShift and multi-cloud monitoring.

Kubernetes and container monitoring

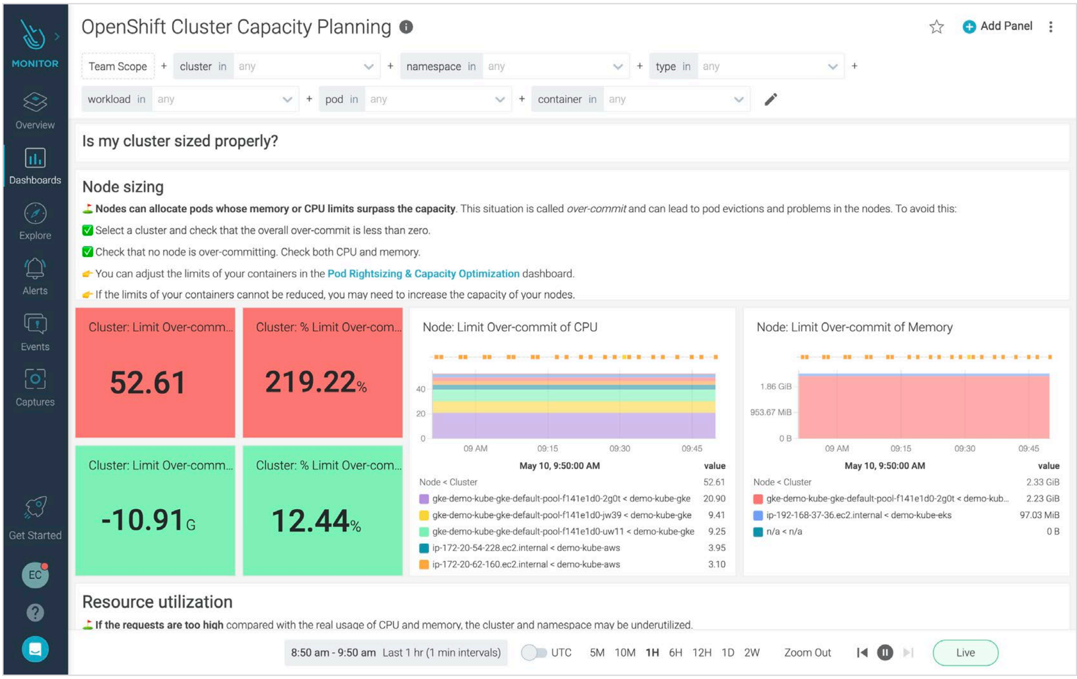
With Sysdig, cloud teams receive automatic alerts and detailed health and performance information, including golden signals for clusters, deployments, namespaces, and workloads. Deep visibility into container activity enriched with cloud and Kubernetes context allows teams to manage the complexity of container deployments. This allows you to:

- Monitor health and performance with deep visibility into infrastructure, services, and applications.
- Visualize the operational status of your clusters with Kubernetes orchestration context.
- Immediately identify owners for issue resolution using container and cloud context.
- Identify pods consuming excessive resources and monitor capacity limits.
- Monitor application auto-scaling behavior to control unexpected billing.
- Reduce cost by optimizing capacity across clusters and clouds.


MONITOR
Overview
Dashboards
Explore
Alerts
Events
Captures
Get Started
EC
OpenShift cluster dashboard
Add Panel
Team Scope
No. of Deployments
93
No change compared to 1 week ago
No. of Replica Sets
92 ▼
0 compared to 1 week ago
No. of Pods
168 ▲
+4 compared to 1 week ago
No. of Services
100
No change compared to 1 week ago
No. of nodes
10
No change compared to 1 week ago
No. of Containers
208 ▲
+10 compared to 1 week ago
Containers per Hosts
20
0
08:30
08:45
09 AM
09:15
May 10, 9:20:00 AM value
sum(timeAvg(container.co...
gke-demo-kube-... 36
gke-demo-kube-... 32
gke-demo-kube-... 31
ip-172-20-54-228 24
ip-172-20-62-160 24
ip-172-20-46-222 23
Response Time by Service
1 s
500 ms
0 ns
08:30
08:45
09 AM
09:15
May 10, 9:20:00 value
avg(avg(net.http.request.ti...
gke-demo-... 1.18 s
gke-demo-... 1.04 s
gke-demo-... 109.82 ms
ip-172-20-4... 11.19 ms
ip-192-168-... 6.61 ms
ip-172-20-5... 6.39 ms
Slowest Resources
metadata.google.internal/computeMetadata/v1/instance/network-interfaces/
carts/carts/606c6ecd4ed9200001dee189/items
front-end/cart/819e1fbf-8b7e-4f6d-811f-693534916a8b
carts/carts/606c6ecd4ed9200001dee189/items/819e1fbf-8b7e-4f6d-811f-693534916a8b
metadata.google.internal/computeMetadata/v1/
metadata.google.internal/computeMetadata/v1/instance/virtual-clock/drift-token
carts/carts/606c6ecd4ed9200001dee186/items
front-end/cart
carts/carts/606c6ecd4ed9200001dee18c/items
front-end/cart/a0a4f044-b040-410d-8ead-4de0446aec7e
0 ns
10 s
20 s
30 s
40 s
50 s
1 m
1.17 m
Top Namespaces (container count)
kube-system
sysdig-agent
monitoring
sock-shop
istio-bookinfo
istio-system
istio-flasknginx
example-voting-app
logdna-agent
example-java-app
0
10
20
30
40
50
Host Capacity Utilization
Top Resources (request count)
8:20 am - 9:20 am Last 1 hr (1 min intervals)
UTC
5M
10M
1H
6H
12H
1D
2W
Zoom Out
Live

With Sysdig, you'll be able to better manage the capacity of your OpenShift clusters, troubleshoot problematic applications and workloads, and rightsize your pods to ensure they meet your requirements. Using Sysdig Monitor, you don't need to be a Kubernetes expert to increase the performance and availability of your clusters.

Sysdig's Kubernetes best practices dashboards provide a more intuitive way of monitoring clusters, containers, and cloud. Our dashboards are designed to be use-case-oriented, providing a well structured, opinionated guide for your OpenShift cluster. Our tips will help you troubleshoot problems faster, understand why issues are happening, and quickly get to the root cause.



OpenShift Cluster Capacity Planning

Is my cluster sized properly?

Node sizing

Nodes can allocate pods whose memory or CPU limits surpass the capacity. This situation is called *over-commit* and can lead to pod evictions and problems in the nodes. To avoid this:

✅ Select a cluster and check that the overall over-commit is less than zero.

✅ Check that no node is over-committing. Check both CPU and memory.

👉 You can adjust the limits of your containers in the **Pod Rightsizing & Capacity Optimization** dashboard.

👉 If the limits of your containers cannot be reduced, you may need to increase the capacity of your nodes.

Node: Limit Over-commit of CPU

May 10, 9:50:00 AM	value
Node < Cluster	52.61
🟪 gke-demo-kube-gke-default-pool-f141e1d0-2g0t < demo-kube-gke	20.90
🟨 gke-demo-kube-gke-default-pool-f141e1d0-jw39 < demo-kube-gke	9.41
🟩 gke-demo-kube-gke-default-pool-f141e1d0-uw11 < demo-kube-gke	9.25
🟦 ip-172-20-54-228.ec2.internal < demo-kube-aws	3.95
🟧 ip-172-20-62-160.ec2.internal < demo-kube-aws	3.10

Node: Limit Over-commit of Memory

May 10, 9:50:00 AM	value
Node < Cluster	2.33 GiB
🟥 gke-demo-kube-gke-default-pool-f141e1d0-2g0t < demo-kub...	2.23 GiB
🟦 ip-192-168-37-36.ec2.internal < demo-kube-eks	97.03 MiB
🟦 n/a < n/a	0 B

Resource utilization

If the requests are too high compared with the real usage of CPU and memory, the cluster and namespace may be underutilized.

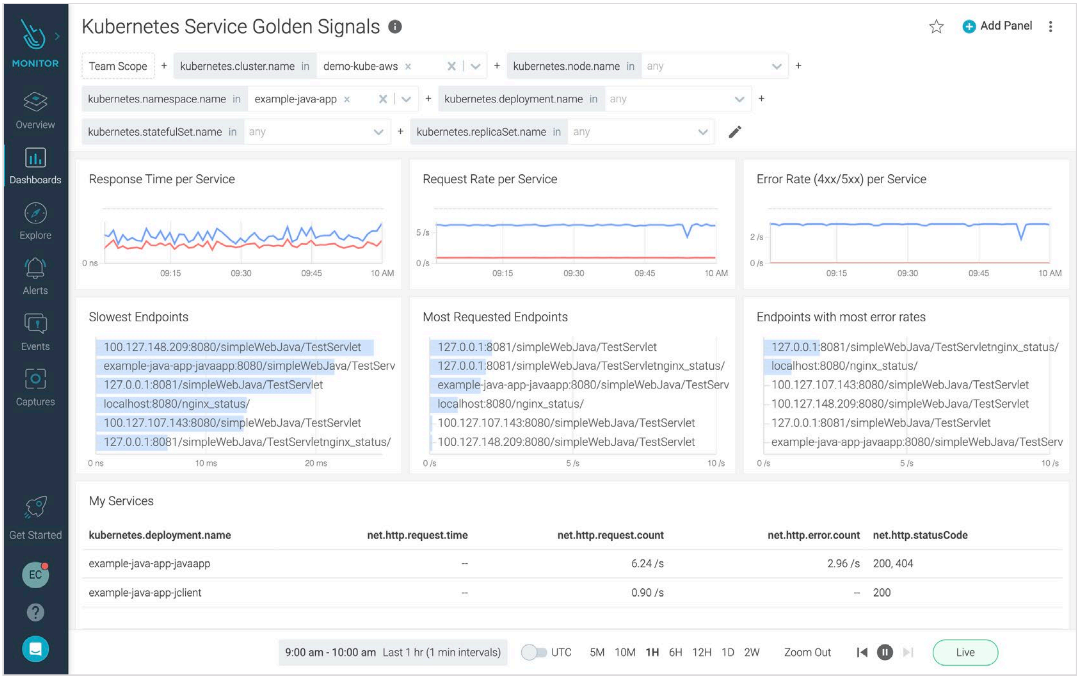
Application and services monitoring

Latency, error, traffic, and saturation metrics are known as the golden signals for monitoring service health. These metrics indicate the real health and performance of your application as seen by users interacting with that service. You can save time by looking at what really matters and avoiding traps that mask the actual problems with applications.

Sysdig Monitor enables you to:

- Accelerate time to insight, with a single source of truth for application availability and security, so teams can resolve issues faster.
- Improve application performance and rapidly solve issues with deep container visibility and granular metrics enriched with Kubernetes and cloud context.
- Observe metrics from cloud services, databases, and other key components in your OpenShift environment using out-of-the-box dashboards.
- Monitor the impact of a given security incident on the availability of a service to your users.
- Reduce risk by utilizing enterprise-grade access controls for your monitoring system, including teams, SSO, and RBAC.
- Leverage your existing developer investment with full Prometheus and PromQL compatibility at cloud-scale.
- Extend monitoring to hundreds of applications and services using Prometheus compatible exporters, dashboards, and alerts.

- Get productive faster by using curated, documented, and supported monitoring integrations for Kubernetes platforms and cloud-native services available from PromCat.io.


MONITOR
Overview
Dashboards
Explore
Alerts
Events
Captures
Get Started
EC
Add Panel
Response Time per Service
0 ns
09:15
09:30
09:45
10 AM
Request Rate per Service
5 /s
0 /s
09:15
09:30
09:45
10 AM
Error Rate (4xx/5xx) per Service
2 /s
0 /s
09:15
09:30
09:45
10 AM
Slowest Endpoints
100.127.148.209:8080/simpleWebJava/TestServlet
example-java-app-javaapp:8080/simpleWebJava/TestServ
127.0.0.1:8081/simpleWebJava/TestServlet
localhost:8080/nginx_status/
100.127.107.143:8080/simpleWebJava/TestServlet
127.0.0.1:8081/simpleWebJava/TestServletnginx_status/
0 ns
10 ms
20 ms
Most Requested Endpoints
127.0.0.1:8081/simpleWebJava/TestServlet
127.0.0.1:8081/simpleWebJava/TestServletnginx_status/
example-java-app-javaapp:8080/simpleWebJava/TestServ
localhost:8080/nginx_status/
100.127.107.143:8080/simpleWebJava/TestServlet
100.127.148.209:8080/simpleWebJava/TestServlet
0 /s
5 /s
10 /s
Endpoints with most error rates
127.0.0.1:8081/simpleWebJava/TestServletnginx_status/
localhost:8080/nginx_status/
100.127.107.143:8080/simpleWebJava/TestServlet
100.127.148.209:8080/simpleWebJava/TestServlet
127.0.0.1:8081/simpleWebJava/TestServlet
example-java-app-javaapp:8080/simpleWebJava/TestServ
0 /s
5 /s
10 /s
UTC
5M
10M
1H
6H
12H
1D
2W
Zoom Out
Live

Kubernetes Service Golden Signals

Team Scope + kubernetes.cluster.name in demo-kube-aws + kubernetes.node.name in any +
kubernetes.namespace.name in example-java-app + kubernetes.deployment.name in any +
kubernetes.statefulSet.name in any + kubernetes.replicaSet.name in any

My Services

kubernetes.deployment.name	net.http.request.time	net.http.request.count	net.http.error.count	net.http.statusCode
example-java-app-javaapp	--	6.24 /s	2.96 /s	200, 404
example-java-app-jclient	--	0.90 /s	--	200

9:00 am - 10:00 am Last 1 hr (1 min intervals)

Container Forensics and Incident Response

When troubleshooting an issue or performing a post-mortem analysis of a security incident involving containers, one of the challenges you'll face is that when a container is destroyed, so is the relevant information it contained.

With container solutions like OpenShift, this happens all the time. Containers run across nodes and services are scaled up and down, starting and deleting container instances. The ephemeral nature of containers makes it difficult to analyze what happened with a security incident after the container is gone. How can you reproduce the steps taken by the intruder? How did they gain access? What was the impact? You need activity data to be able to identify the root cause of problems and recognize whether the issue comes from malicious activity or a misconfiguration of the app.

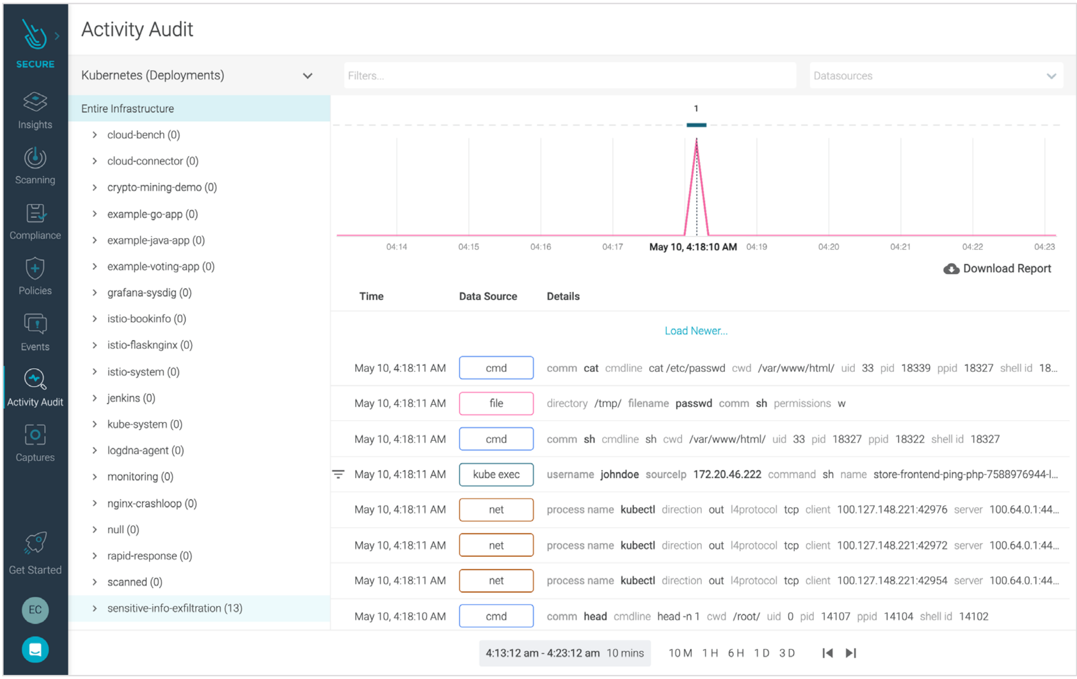
Red Hat provides...

OpenShift Logging aggregates and stores from your cluster, such as node system audit logs, application container logs, and infrastructure logs to use for debugging. event corroboration, and similar purposes. Cluster administrators can view all logs and application developers can analyze logs for projects they have permission to view.

With OpenShift you can use a Kibana web console to visualize log data. In addition, you send logs to other log aggregators using the OpenShift Container Platform Log Forwarding API.

Sysdig adds...

Sysdig's Activity Audit speeds incident response and enables audit for OpenShift clusters. Sysdig captures and correlates executed commands, network, and orchestrator activity so Security Operations Center (SOC) teams can spot what happened. With Sysdig captures, you can also record all container activity at a detailed level, including spawned processes, network connections, file system activity, etc., so you can understand events in detail and conduct forensics even after the container is gone.



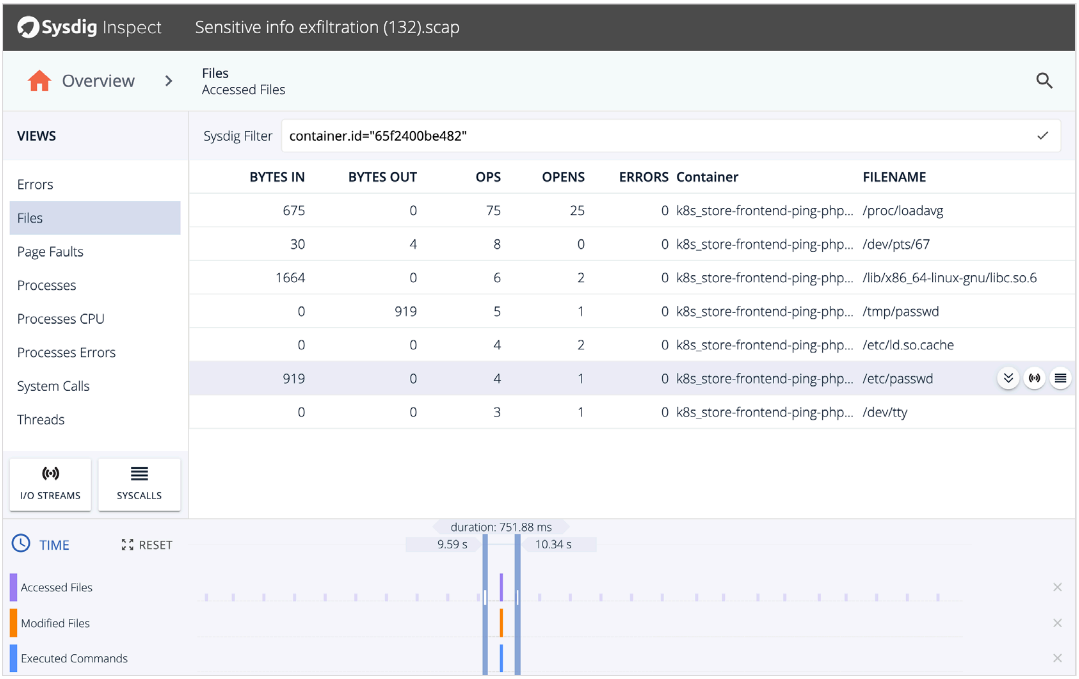
Time	Data Source	Details
May 10, 4:18:11 AM	cmd	comm **cat** cmdline cat /etc/passwd cwd /var/www/html/ uid 33 pid 18339 ppid 18327 shell id 18...
May 10, 4:18:11 AM	file	directory /tmp/ filename **passwd** comm **sh** permissions w
May 10, 4:18:11 AM	cmd	comm **sh** cmdline sh cwd /var/www/html/ uid 33 pid 18327 ppid 18322 shell id 18327
May 10, 4:18:11 AM	kube exec	username **johndoe** sourceIp 172.20.46.222 command sh name store-frontend-ping-php-7588976944-l...
May 10, 4:18:11 AM	net	process name **kubectl** direction out l4protocol tcp client 100.127.148.221:42976 server 100.64.0.1:44...
May 10, 4:18:11 AM	net	process name **kubectl** direction out l4protocol tcp client 100.127.148.221:42972 server 100.64.0.1:44...
May 10, 4:18:11 AM	net	process name **kubectl** direction out l4protocol tcp client 100.127.148.221:42954 server 100.64.0.1:44...
May 10, 4:18:10 AM	cmd	comm **head** cmdline head -n 1 cwd /root/ uid 0 pid 14107 ppid 14104 shell id 14102

Read more about Sysdig audit capabilities in Incident response in Kubernetes with Sysdig's Activity Audit. Notifications sent to your alerting channels or SIEM allow you to stay on top of activity in your OpenShift clusters and help you consolidate security findings across your container environments, letting you view and manage security alerts, as well as automate compliance checks.

Using Sysdig captures, security teams can resolve issues inside pods and conduct forensics by reconstructing system activities. Detailed forensics reports help you quickly understand and contain the impact of any security breach. Using this information, you can:

- Easily recreate the steps taken on intrusion to streamline incident response.
- See file activity, network traffic, application protocols, commands, logs, and events.
- Investigate incidents such as data exfiltration and lateral movement.

- Perform post-mortem analysis on a container outside production.
- Recover quickly and strengthen defenses going forward.



Sysdig Filter: container.id="65f2400be482"

BYTES IN	BYTES OUT	OPS	OPENS	ERRORS	Container	FILENAME
675	0	75	25	0	k8s_store-frontend-ping-php...	/proc/loadavg
30	4	8	0	0	k8s_store-frontend-ping-php...	/dev/pts/67
1664	0	6	2	0	k8s_store-frontend-ping-php...	/lib/x86_64-linux-gnu/libc.so.6
0	919	5	1	0	k8s_store-frontend-ping-php...	/tmp/passwd
0	0	4	2	0	k8s_store-frontend-ping-php...	/etc/ld.so.cache
919	0	4	1	0	k8s_store-frontend-ping-php...	/etc/passwd
0	0	3	1	0	k8s_store-frontend-ping-php...	/dev/tty

Red Hat OpenShift + Sysdig: Better Together

Sysdig and Red Hat have a long standing partnership established to help customers confidently migrate and develop applications to run on top of OpenShift and cloud. Red Hat provides a secure-by-default container platform with OpenShift. The Sysdig Secure DevOps Platform complements OpenShift to give you comprehensive security and monitoring visibility to understand the behavior of your clusters, containers, and cloud. Together, Sysdig and Red Hat enable you to securely run container workloads, meet stringent compliance requirements, and observe performance and availability.

The following table summarizes the security and monitoring capabilities highlighted in this guide, along with the benefits of using the Sysdig Secure DevOps Platform to complement Red Hat solutions for security, compliance, and monitoring.

Security & Compliance

Security Layers	OpenShift	Benefit of Sysdig + OpenShift
Host OS Security	RHEL/RHCOS	Continuously scan the underlying container host configurations and ensure it meets CIS benchmarks.
Access Control	OAuth/RBAC	Implement service-based access control to streamline security and monitoring information to an individual user or team.
Image Scanning	Clair (Package Image scanning)	Scan images pre-deployment within your CI/CD pipelines or any OpenShift registries (Quay, DockerHub, ECR, etc.) Identify vulnerabilities within OS packages, third-party libraries, Dockerfile, and more. Get runtime reporting to assess the impact of new CVEs.

Compliance	OpenSCAP Red Hat Insights	Continuously validate compliance using out-of-the-box image scanning policies, automated CIS benchmark checks, container runtime policies, and compliance dashboards.
Network Security	Software-defined networking (SDN) Kubernetes network policies	Automate and simplify use of native Kubernetes network policies. Visualize all network communication between pods, services, and applications. Audit connections to or from any process and implement a Zero Trust approach to container security.
File Integrity Monitoring		Use runtime filesystem policies to quickly implement file integrity monitoring (FIM) and alert on any suspicious changes to files and directories.
Cloud Workload Protection Runtime Detection & Threat Prevention		Scope runtime security policies using OpenShift/Kubernetes labels and metadata to detect and prevent anomalous behavior. Detect and block attacks, combining deep visibility through system calls, logs, and audit events. Powered by open-source CNCF runtime security project Falco.
Cloud Security Posture Management		Discover cloud assets, gain visibility into configuration issues, and detect cloud service threats. Unify CSPM and cloud threat detection with container security to reduce the time to detect threats for OpenShift on public clouds.
Container Forensics	OpenShift Logging	Conduct forensics and post-mortem analysis in-depth with system call captures of activity before, during, and after an event.

Monitoring

OpenShift and Kubernetes cluster monitoring Container, workload, and cloud service monitoring	Prometheus	Identify and resolve issues faster. Get deep visibility into OpenShift container infrastructure, services, and applications. Visualize and correlate containers, Kubernetes, and infrastructure metrics and events across clusters and clouds. Extend Prometheus with scale and enterprise features without sacrificing compatibility. Monitor hundreds of applications and services using Prometheus-compatible exporters, out-of-the-box dashboards, and alerts.
Troubleshooting	OpenShift Logging	Get powerful kernel-level observability to troubleshoot host, network, application, container, and process issues – even after OpenShift terminates containers/ pods.

Conclusion

Red Hat OpenShift is helping enterprises move fast and innovate to deliver solutions that meet customer and market needs. Red Hat provides a secure-by-design platform and monitoring for your container infrastructure. As you scale out your applications, clusters, locations, and integrations, Sysdig helps you confidently run containers, Kubernetes, and cloud with a container and cloud security stack built on open source.

Sysdig complements Red Hat solutions with deep, unified security and visibility. Our platform is radically simple to run and scale, protecting your workloads wherever you choose to operate OpenShift to support your business applications.

Additional Resources

Get Started Free

Sysdig Secure DevOps Platform Free Trial

Partnership Overview

Red Hat + Sysdig Partner Page

Red Hat + Sysdig Partner Brief

Case Studies

Worldpay Gains Competitive Edge with PCI-Compliant Payment Solutions

Ford Motor Company Optimizes Delivery with Cloud Platform

Videos

Ford Optimizes Container Platform to Speed Delivery

ATPCO Secures and Monitors Industry Flight-shopping Services

Sysdig and Red Hat Partnership: Visibility and Security

Webinars

Solving Kubernetes security issues using Red Hat OpenShift & Sysdig

Fighting Fraud: Worldpay Protects Cardholder Data with Sysdig & Red Hat OpenShift

Consistent Container Vulnerability Scanning with Red Hat & Sysdig

Find out how the Sysdig Secure DevOps Platform can help you and your teams confidently run cloud-native apps in production. Contact us for additional details about the platform, or to arrange a personalized demo.

www.sysdig.com

sysdig