5 Keys to Securing AWS Cloud and Containers

As container and cloud adoption accelerates, enterprises struggle with visibility into their container and cloud environments. According to Gartner, "Nearly all successful attacks on cloud services are the result of customer misconfiguration and mistakes.[1]

Containers are essentially black boxes. It's hard to see what's going on inside. Adding to this challenge, container lifespans are very short. According to the Sysdig 2022 Cloud-Native Security and Usage Report, 44% of containers live less than five minutes. Traditional security tools can't see inside containers or provide insight into short-lived workloads in a dynamic Kubernetes deployment.

How can you automate efficient security and compliance controls for containers and cloud services in your Amazon Web Services (AWS) environment? Do you have the visibility and security control required to successfully run workloads on with Amazon EKS, Amazon ECS, AWS Fargate? With the right set of integrated tools built for a cloud-native environment, you can successfully manage cloud and container security risk for all your AWS accounts, infrastructure, and workloads

The key is to reduce your risk from cloud misconfigurations, continuously scan for cloud and container vulnerabilities, detect abnormal activity, and prioritize threats to ensure your applications are secure across their entire life cycle. These five key workflows will enable you to cover the most critical security and visibility requirements so you can confidently and securely run containers, Kubernetes, and cloud operations in AWS.

[1] Gartner: Innovation Insight for Cloud Security Posture Management

1

5 KEYS TO SECURING AWS CLOUD AND CONTAINERS

Enable Continuous Cloud Security

Continuous cloud security is necessary to immediately identify configuration errors and suspicious behavior. In a shared responsibility model, it is the job of AWS users to implement and manage these safeguards. The following steps can help you validate your cloud security posture.

Automatically discover the assets running in your AWS environment including systems, applications, and services like Amazon VPC, Amazon RDS, Amazon S3, Amazon ECS, Amazon EKS, and AWS Fargate.

Scan infrastructure-as-code templates for adherence to security policies and auto-remediate at the source with a Git pull request (PR).

Check your cloud configurations periodically against Center for Internet Security (CIS) benchmarks to identify misconfigurations (e.g., public storage buckets, exposed security groups and access controls, etc.) and take steps to remediate violations.

Ensure visibility into user and service access to cloud resources with Cloud Infrastructure Entitlements Management (CIEM) to remediate excessive permissions.

Monitor activity across your cloud accounts, users, and services using AWS CloudTrail logs and policies that alert on service changes, suspicious behavior, and potential threats.

Automate compliance and governance by consistently applying policies across the application lifecycle.

2

5 KEYS TO SECURING AWS CLOUD AND CONTAINERS

Manage Vulnerabilities: Scan container images and hosts

As your container images, versions, and builds proliferate, you can lose control of what software is being used and whether required software updates are applied. "Shift left" security practices focus on embedding security into your delivery pipeline. By identifying and prioritizing vulnerabilities based on real risk you can address issues earlier, so you don't slow down deployment.

Embed scanning into registries like Amazon ECR and CI/CD pipelines like AWS CodePipeline to prevent risky images from being deployed.

Adopt in-line scanning that checks images within your cloud account to maintain full control of your images.

Automate image scanning for AWS Fargate serverless tasks to reduce the risk of running vulnerable containers.

Validate the build configuration such as Dockerfile instructions and image attributes like size and labels to align with your security policies.

Identify newly reported vulnerabilities that impact a container already running in production.

Set unique policies for images from public repositories versus images built in-house. Consider strict checks for images pulled from external sources.

Leverage Kubernetes and cloud context to identify and alert the right team to address issues.

Notify developer teams directly in their CI/CD tooling to streamline addressing vulnerabilities.

Scan Amazon EC2 instances to identify known vulnerabilities in OS and non-OS packages.

3

5 KEYS TO SECURING AWS CLOUD AND CONTAINERS

Detect and respond to runtime threats

Even after ensuring the right configurations and addressing vulnerabilities, you need to be vigilant in monitoring runtime activity. Zero-day attacks and malicious actors can threaten your production applications and data. Reducing runtime risk requires threat detection and response built for cloud and containers.

Implement runtime security to observe workload behavior, monitor cloud activity, and identify anomalous events across AWS cloud and container services.

- → Observe the behavior of your containers using a trusted source of truth such as Linux system calls.
- → Create and maintain runtime security policies that observe workload behavior, monitor cloud activity, and identify anomalous events.
- → Apply security policies based on container role and Kubernetes context.
- → Automate AWS CloudTrail log filtering to detect unexpected service activity and configuration changes.

For serverless applications on AWS Fargate, automate instrumentation to ensure visibility into threats in real-time, even for short-lived tasks.

Leverage Kubernetes-native controls available with Amazon EKS for runtime protection of container workloads.

- → Use Admission Controllers to define what is allowed to run on your clusters.
- → Enforce “least privilege” for clusters and containers, providing only the permissions needed to perform required tasks.
- → Implement least-privilege Kubernetes network policies with Amazon EKS.

Monitor for container drift and block execution of packages or binary files added or modified after a container is deployed into production.

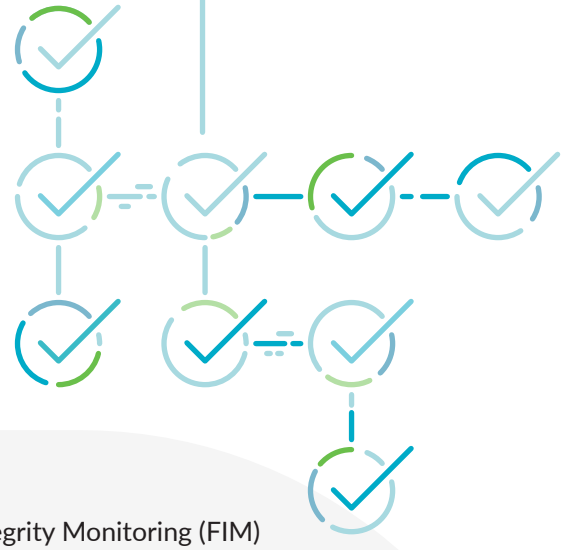
Capture detailed activity records to streamline incident response and quickly respond to container and cloud security threats even after containers are gone.

4

5 KEYS TO SECURING AWS CLOUD AND CONTAINERS

Continuously validate compliance

Your organization likely needs to meet certain compliance mandates such as SOC2, PCI, NIST, HIPAA, etc. Fast-moving cloud environments present unique challenges for monitoring and measuring compliance. Solutions area available to help simplify the job of regularly checking your environment to ensure you're meeting compliance best practices.



Check your container and platform configuration against CIS benchmarks for AWS, Docker, and Kubernetes.

Validate compliance during build-time by mapping container image scanning policies to standards (e.g., NIST, PCI, SOC2, or HIPAA) or internal compliance policies (e.g., blacklisted images, packages, or licenses).

Manage compliance at runtime using policies tailored to regulatory standards that ensure you're following best practices (e.g., don't run privileged containers or run containers as root).

Implement policies that specifically look for known adversary tactics, techniques, and procedures (TTPs).

Implement File integrity Monitoring (FIM) to detect tampering of critical system files, directories, and unauthorized data changes.

Capture activity audits to track network, file, command, and kube-api activity for proof-of-compliance with required controls.

Provide proof of container compliance with vulnerability scanning reports.

Capture detailed activity records surrounding incidents to use for forensics and investigations as needed.

Monitor configuration and policy changes across your AWS cloud services using AWS CloudTrail.

5

5 KEYS TO SECURING AWS CLOUD AND CONTAINERS

Monitor and troubleshoot containers, Kubernetes and cloud

Containers and cloud services are dynamic and churn constantly. Visibility into the health and performance of your AWS workloads and infrastructure is critical for ensuring the availability of your cloud applications.

Implement monitoring built for containers, Kubernetes, and AWS cloud services. To improve application performance and rapidly solve issues, you need deep visibility and granular metrics enriched with Kubernetes and cloud context.

- → Identify services and workloads consuming excessive resources and monitor capacity limits.
- → Identify owners for issue resolution using container and cloud context.

Tap into standards for cloud monitoring including open-source Prometheus and AWS CloudWatch. Combining these two vantage points help you achieve observability for services like AWS Fargate, Amazon S3, Amazon RDS, and AWS Lambda.

- → Get productive quickly with curated, documented, and supported monitoring integrations.

Monitor the Kubernetes state to understand the health of your container orchestration service.

- → Monitor key metrics for nodes, namespaces, pods, etc. to identify errors and ensure you have sufficient resources to run applications.
- → Reduce cost by optimizing capacity across clusters and clouds.

Capture historical data and detailed system activity to ensure you can investigate, correlate, and resolve issues quickly.

Monitor CPU and resource usage as indicators of attacks such as DoS and crypto-mining. Gain insights into attacks and spread vectors by monitoring network connections.

Secure Your Cloud and Containers

www.sysdig.com

START YOUR FREE TRIAL

REQUEST PERSONALIZED DEMO