

Licensing Guide
October' 24

Kaspersky Container Security

kaspersky bring on
the future

Part of



Kaspersky
Cloud Workload
Security



Kaspersky Container Security

Containerization

is one of the primary global software development trends right now. Most companies globally use containers in their apps. The technology shortens time-to-market, enables more rational use of computer resources, and delivers robust and well-built apps to customers. However, the architectural features of containerized apps prevent traditional and open-source solutions designed for code analysis and endpoint protection from providing adequate information security.

Kaspersky Container Security (KCS) protects every stage of a containerized app's lifecycle, from development to operation. It protects your organization's business processes in line with security standards and regulations, and supports implementation DevSecOps.

Kaspersky Container Security delivers comprehensive protection from the latest cyberthreats. It automates your compliance audits, freeing up your security team's resources so they can focus on other tasks, and shortens time-to-market.

Kaspersky Container Security has been developed specifically for on-premise and cloud containerized environments, ensuring protection at different levels, from container image to host OS.

Kaspersky Container Security is a part of the Kaspersky Cloud Workload Security solution. It provides comprehensive protection from attacks and reduces threat detection and response times in cloud environments.

Licensing levels



Kaspersky Container Security

Standard

Provides container image protection, integration with image registries, orchestrators, CI/CD platforms, and SIEM solutions



Kaspersky Container Security

Advanced

Ensures protection of containers in the runtime environment, provides enhanced monitoring capabilities and tools for compliance checks

Features and licensing tiers

Features

Standard Advanced

Integration with container image registries

Integrates with Docker Hub, JFrog, Sonatype Nexus OSS, GitLab Registry, Harbor



Orchestration environment support

Supports Kubernetes and OpenShift



Integration with public clouds

Supports AWS and Microsoft Azure



Scanning of images for malicious objects, vulnerabilities and secrets

Scanning can be performed manually or automatically based on predefined parameters



Risk assessment for container images and configuration files (IaC)

Automated image assessment based on criticality levels



Scanning of configuration files (IaC)

Configuration error detection and best practice checks



Set of criteria in UI for creating custom policies and editing preset policies

Enables creation of policies for image security scanning, response, and runtime analysis



Integration with CI/CD platforms and scanning of images and IaC at development stage

Integrates with Jenkins, Team City and Circle CI to block images and containers when security threats are detected



Visualization tools

Visualization of information about images, containers, and infrastructure elements



Reporting system

Generation of reports and ability to download them from the log on demand



Integration with external security and notification systems

Integration with SIEM (via syslog), LDAP, e-mail, Telegram



Open API for key product functionality (Swagger)

Integration and installation convenience improvement



Container launch monitoring and control in accordance with security policies

Product can prohibit launch of non-compliant images, unregistered images, and images with privileges, as well as mount specific datastores in containers.



Detecting and scanning images in a cluster

Ability to scan images at runtime



Behavioral analytics of containers (based on templates)

Monitoring containers based on the preset profile



Container integrity monitoring

Monitoring consistency between scanned image and image from which container is running



File threat protection for running containers (eBPF and KESL -based)

Preventing potential attacks on orchestrator via containers in runtime



Controls the launch of applications and services inside containers

Detecting and blocking suspicious activity inside containers



Monitors the traffic of running containers

Detecting and blocking suspicious activity between containers in cluster and between clusters



Container platform component configuration analysis for best practice and regulatory compliance

Infrastructure analysis for compliance with best protection practices to improve environment security level



Visualization of resources in a cluster

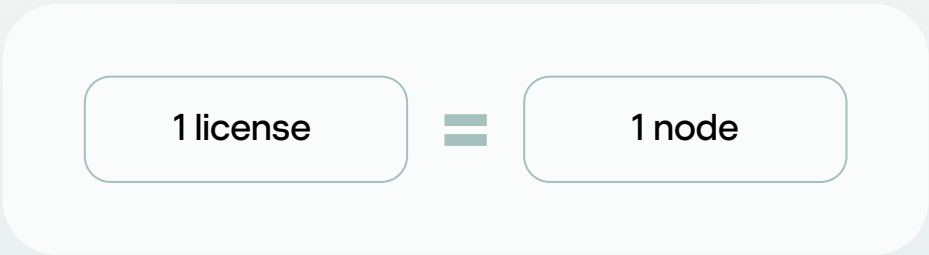
View key information about the state of a cluster and its components



Licensing objects

Nodes with containers

Quantity of nodes on which the KSC Agent is deployed are taken into account



Premium technical support

Kaspersky Premium support is provided within **Kaspersky Maintenance Service Agreement (MSA)** and focused on superior user experience with high class priority maintenance. For Kaspersky Container Security you may choose out of two support options: MSA Business for KCS or MSA Enterprise for KCS.

	MSA Business for KCS	MSA Enterprise for KCS
Request receiving availability	Criticality level 1 — on 24×7, the rest — standard office hours of the Kaspersky Local Office	Criticality level 1 and 2 — on 24×7, the rest — standard office hours of the Kaspersky Local Office
Response time	Criticality level 1 — 2 hours* Criticality level 2 — 6 business hours Criticality level 3 — 8 business hours Criticality level 4 — 10 business hours	Criticality level 1 — 30 minutes* Criticality level 2 — 4 hours* Criticality level 3 — 6 business hours Criticality level 4 — 8 business hours
Contact persons	4 — the possible number of contact persons from the customer's side	8 — the possible number of contact persons from the customer's side
		Dedicated Technical Account Manager (TAM) Provides reports to the customer on open incidents

⌚ Note: Please check availability of MSA contracts and all terms and conditions in your country with your account manager

* Outside of business hours, additional contact by phone is required

License calculation examples

Scenario A

The customer needs to secure container images ONLY

Scenario B

The customer needs to secure not only container images, but also runtime apps, and they also want to check their compliance

For example, in both cases the customer has a total of 810 nodes deployed in infrastructure. On 500 nodes from total amount deployment of containers is planned. Despite the customer purposes described in scenarios A and B we should consider only nodes on which containers are deployed where 1 node count as a 1 license.

500 nodes = 500 licenses

500 licenses

Kaspersky Container Security Standard

500 licenses

Kaspersky Container Security Advanced

Advantages for business



Globally renowned security

- Kaspersky Container Security's features and capabilities are in line with global best practices for container security
- Internationally recognized and award-winning protection



Easy operation – reliable protection

- Real-time visualization of threats
- Reduces the necessity of involving the information security team while improving the quality and speed of security checks



Comprehensive protection for containerized environments

- Protection at different levels of the containerized environment architecture
- App security for every stage of the lifecycle

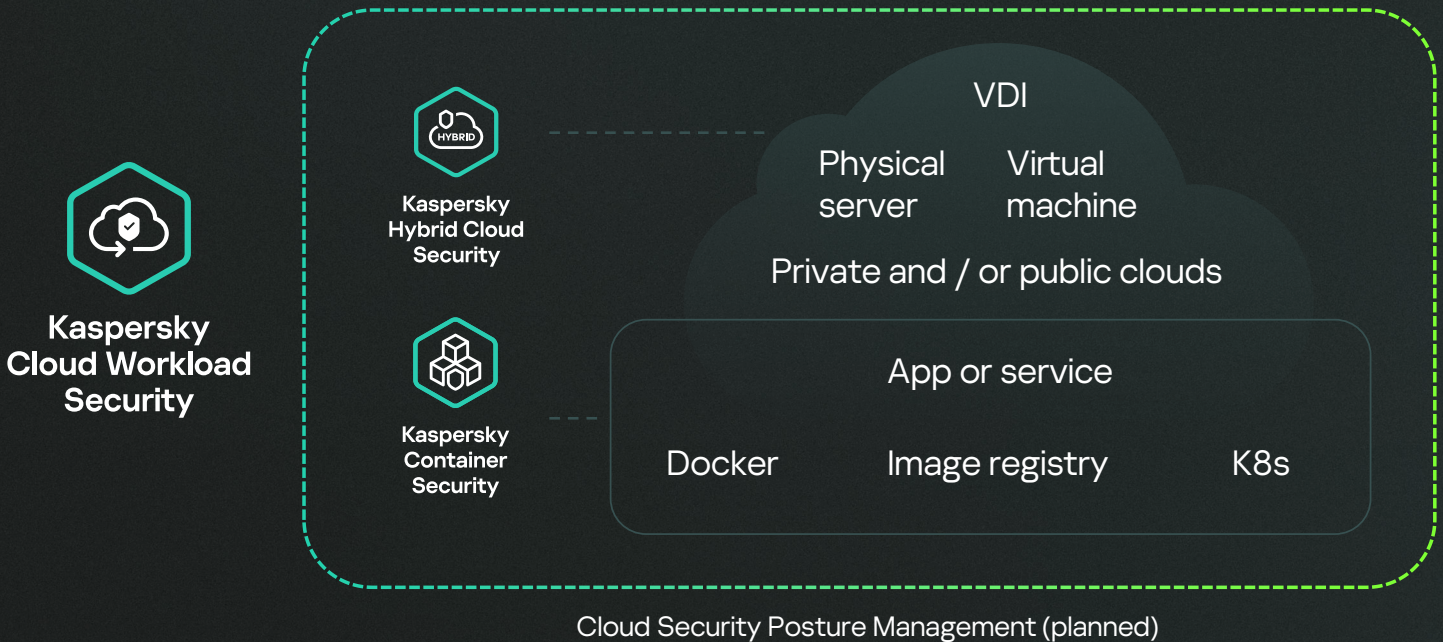


Regulatory compliance

- Best practices audits
- Transparent reporting system

Part of Kaspersky Cloud Workload Security

Kaspersky Container Security in combination with Kaspersky Hybrid Cloud Security forms a comprehensive cloud workload security solution for reliable, world-class protection from attacks together with shorter threat detection and response times in cloud environments. The Kaspersky Cloud Workload Security solution ensures comprehensive protection of your hybrid and cloud infrastructures: virtual machines / container clusters.



Supported solutions





Kaspersky Container Security

[Learn more](#)

www.kaspersky.com

© 2024 AO Kaspersky Lab.
Registered trademarks and service marks
are the property of their respective owners.

[#kaspersky](#)
[#bringonthefuture](#)