DevOpsSchool

Course curriculum · November 2026

LEVEL 04 OF 04 · ADVANCED

Deep Dive into Docker Security

Operate a full Docker security toolchain, secure Swarm clusters end to end, and govern container security from build to incident response.

LIVE ONLINE **Max 8-10 hours**	CORPORATE (DAY-WISE) **5 days · Days 1-5**	SELF-PACED VIDEO **1-year LMS access**
BATCH SIZE **Max 10 learners**	NEXT BATCHES **1st of every month · weekends: first Saturday**	CERTIFICATE **Verifiable certificate of completion**

Foundation	Essential	Intermediate	Advanced
1 day · Included	2 days · Included	3 days · Included	5 days · This curriculum

INSIDE THIS CURRICULUM

What you will achieve	Outcomes and level focus
Who should attend	Audience and prerequisites
Agenda	51 topics · Foundation to Advanced
Hands-on labs	4 labs
Formats and fees	Live, 1:1, self-paced, corporate
Trainer and reviews	Rajesh Kumar
Enrolment	Batches, payment, contacts

WHY DEVOPSSCHOOL

- Rated 4.6 on Google
- 1,500+ verifiable certificates issued in the public registry
- Trainer Rajesh Kumar: 20 years in industry
- Small batches: max 10 learners, every session recorded

Course page: www.devopsschool.com/courses/docker/docker-security-training-advance.html
Talk to us: WhatsApp +91 99057 40781 · +1 (469) 756-6329 · contact@DevOpsSchool.com · reply within 1 hour

OUTCOMES

What you will achieve

By the end of the Deep Dive into Docker Security course path, you will be able to:

1

Harden hosts and daemons

Baseline and harden Docker hosts and daemons against the CIS Docker Benchmark with audit rules, TLS and user namespaces.

2

Secure the image supply chain

Run authenticated private registries with scanning and signature verification so only trusted images reach production.

3

Detect and respond at runtime

Monitor running containers with tools such as Falco and respond to breakouts, abuse and leaked secrets.

WHAT THE ADVANCED LEVEL ADDS

Security tooling, Swarm security and governance

Hands-on: End-to-end Docker security capstone. Build a CI pipeline that scans and signs images, deploy them to a locked Swarm with secrets, and detect a simulated breakout with Falco.

WHO SHOULD ATTEND

- Experienced Deep Dive into Docker Security practitioners responsible for quality, security and scale.
- Senior engineers, architects and leads who design and troubleshoot Deep Dive into Docker Security solutions.
- Teams making Deep Dive into Docker Security an organisation-wide standard.

BEFORE YOU START

- The Intermediate level, or equivalent hands-on experience.
- You can already: Reduce container breakout risk, enforce fine-grained Docker API authorization with OPA, and run a TLS-secured, authenticated private registry.
- Basic computer skills and a laptop with a stable internet connection for the live sessions and labs.
- This curriculum includes every earlier level, so it is complete on its own.

AGENDA · FOUNDATION TO ADVANCED

Deep Dive into Docker Security agenda

51 topics. Each level builds on the one before it, so this curriculum lists every earlier level in full. Days are for corporate (day-wise) delivery; live online batches cover the same agenda in max 8-10 hours.

DAY 1 Foundation: Container threat model and kernel isolation

Included from Foundation

1. Why containers present special security challenges: Docker security challenges, shared kernel, protecting your Docker containers
2. Container security lifecycle: CI/CD and pre-deployment security, static analysis, runtime security and incident response
3. Container privileges: root inside containers, --privileged risks and the Docker hardening standard (CIS Docker Benchmark) overview
4. Kernel namespaces: pid, net, mnt, ipc, uts and user isolation and what they do not protect
5. Control groups: limiting CPU, memory and pids to prevent container resource abuse and noisy neighbours
6. Linux kernel capabilities: the default capability set, --cap-drop ALL, --cap-add and no-new-privileges
7. Docker daemon attack surface: a root daemon, exposed docker.sock and why docker group membership equals root
8. Docker vulnerabilities and threats to battle: Docker host and kernel security, Docker container breakout paths
9. Image threats: container image authenticity, security vulnerabilities present in the static image, untrusted base images
10. Docker credentials and secrets: secrets baked into layers, environment variables, ~/.docker/config.json exposure
11. Container resource abuse and runtime threats: cryptomining, fork bombs and the need for Docker runtime security monitoring
12. Docker Logging: daemon and container logs, logging drivers and shipping logs centrally for security investigations
13. Best practices: trusted minimal base images, non-root USER, read-only root filesystem and no unused ports

DAY 2 Essential: Platform baseline, host hardening and daemon TLS

Included from Essential

14. Establishing a baseline for Docker platform security: defining the Docker platform and determining what needs to be secured
15. Measuring how secure a Docker platform is: scoring findings and acting on Docker platform security vulnerabilities
16. CIS Docker Benchmark compliance: testing a Docker platform with docker-bench-security, reading and remediating results
17. Optimizing the configuration of the Docker host: separate partition for /var/lib/docker, minimal services, kernel updates
18. Employing minimal operating systems: container-optimized OSes; deploying Docker on RancherOS in the cloud (now end of life) and alternatives
19. Hardening the host operating system: patching, SELinux or AppArmor in enforcing mode, SSH and firewall hardening
20. Keeping the Docker Engine current: release channels, upgrade process and tracking Docker and runc CVEs
21. Auditing important Docker artifacts: creating audit rules for the Docker host with auditd for dockerd, docker.sock and /etc/docker
22. Configuring the Docker daemon for security: daemon.json icc false, no-new-privileges, live-restore, userland-proxy, log driver

23. Controlling access to the Docker daemon socket: socket permissions, docker group membership, never bind-mounting docker.sock
24. Using TLS to protect the Docker daemon: configuring TLS for the Docker client and daemon, tlsverify on port 2376, SSH as alternative
25. Demo and labs: run a CIS benchmark, fix the top findings and re-run to measure improvement

DAY 3 **Intermediate: Breakout prevention, authorization and secure registries** Included from Intermediate

26. Minimizing the risk associated with a container breakout: dropped capabilities, seccomp and AppArmor profiles, no --privileged
27. Implementing user namespaces for containers: userns-remap, subuid and subgid mapping and rootless Docker mode
28. Enhancing access control to the Docker platform: the all-or-nothing default access control mechanism and its limits
29. The Docker Plugin API: how authorization plugins intercept Docker API requests and responses
30. Introducing the Open Policy Agent Docker authorization plugin: installing and configuring opa-docker-authz
31. Defining authorization policy with Rego: deny privileged containers, host bind mounts and unapproved images
32. Implementing fine-grained access control to the Docker platform: per-user policies based on TLS client identities
33. Deploying a secure Docker registry: the registry (CNCF Distribution) image, storage backends and enabling the use of insecure registries
34. Securing communication with a self-hosted Docker registry: configuring TLS for the Docker daemon and registry, certs.d trust
35. Registry authentication: controlling access with basic authentication (htpasswd) and token-based authentication
36. Implementing authentication for a self-hosted registry: token service setup, robot accounts and credential helpers
37. Registry security scanning: scanning images on push with Trivy or Clair and blocking vulnerable images
38. Docker Content Trust signature verification: DOCKER_CONTENT_TRUST=1, Notary signing and Cosign as a current alternative

DAYS 4-5 **Advanced: Security tooling, Swarm security and governance** **New at this level**

39. Tools for Docker security: Docker Bench for Security, Clair, Anchore, Dagda and OpenSCAP Workbench scans
40. Runtime and network security tools: Sysdig Falco rules, Cilium network policies, Banyanops Collector, Dockscan and Batten
41. Supply chain and compliance tools: Notary, Grafeas metadata, BlackDuck Docker security and InSpec compliance profiles
42. Managing security in a Docker Swarm cluster: securing communication between cluster nodes with mutual TLS and certificate rotation
43. Using secrets to manage sensitive artifacts: docker secret create, /run/secrets mounts, secret rotation and encrypted overlay networks
44. Autolocking a cluster to protect the encryption key: docker swarm update --autolock and managing unlock keys
45. Managing the availability of a Swarm cluster: manager quorum sizing and recovering from a lost quorum with --force-new-cluster
46. Runtime security monitoring: Falco alerts on shell spawns, unexpected network activity and sensitive file access
47. Incident response for containers: containment, evidence capture with docker diff, export and logs, and recovery
48. Static analysis in CI/CD: Hadolint for Dockerfiles, Trivy and Docker Scout image scans and policy gates that fail builds

49. Image provenance: SBOM generation with Syft or BuildKit attestations, Cosign signing and verification before deploy
50. Security governance: hardening standard as code, scheduled CIS audits, exception handling and vulnerability SLAs
51. Capstone with demos and labs: secure a Docker platform from host baseline through registry, runtime monitoring and Swarm

PRACTICE

Hands-on labs

You build these yourself during the course, guided by the trainer.

FOUNDATION

Map the attack surface of a host

Inspect namespaces, cgroups and capabilities of running containers, then demonstrate why a mounted docker.sock gives full host access.

ESSENTIAL

Harden a Docker host to CIS

Run docker-bench-security on a fresh host, apply daemon.json, auditd and TLS fixes, and show the improved benchmark score.

INTERMEDIATE

Secure registry with signed images

Deploy a private registry with TLS and token authentication, sign images with Docker Content Trust, and block unsigned or privileged runs with an OPA policy.

ADVANCED

End-to-end Docker security capstone

Build a CI pipeline that scans and signs images, deploy them to a locked Swarm with secrets, and detect a simulated breakout with Falco.

HOW YOU CAN LEARN

Formats and fees

The same agenda in four formats. Fees are in Indian rupees as of November 2026; the course page shows them in your currency.

FORMAT	DURATION	WHAT YOU GET	FEE
Live & interactive (online)	Max 8-10 hours	Instructor-led public batch of max 10 learners with demos, labs, Q&A and assignment review. Weekday batches start on the 1st of every month, weekend batches on the first Saturday of every month.	**INR 34,999**
1:1 live training (online)	Max 8-10 hours, one-to-one	The same agenda, taught one-to-one by the trainer. Starts immediately, without waiting for the next batch; sessions planned around your working hours, at your pace.	**INR 99,999**
Self-paced video	1-year LMS access	All four Deep Dive into Docker Security levels, plus 20+ more courses and 50+ tools, at your own pace.	**INR 833/month** billed yearly at INR 9,996
Corporate (day-wise)	5 days for this level	A private batch for your team, on demand. Foundation 1 day, Essential 2, Intermediate 3, Advanced 5 (cumulative).	**Quote on request**

ASSESSMENT AND CERTIFICATE

Your progress is checked through hands-on labs and assignments at every level. On completion you receive a DevOpsSchool certificate of completion with a unique credential ID that anyone can verify at devopsschool.com/certificates.

INCLUDED WITH EVERY LIVE BATCH

- Every live session is recorded and stays in your LMS for a year.
- Can't attend your batch? Move to next month's batch.
- Free demo class before you pay.
- Free 10-15 minute call with an advisor or the trainer to check the course fits you.

YOUR TRAINER

Learn from Rajesh Kumar

Principal Architect, trainer and mentor in DevOps, SRE, DevSecOps and platform engineering, with 20 years across 8+ software MNCs, including Staff DevOps/SRE Engineer for a leading Japanese mobile-payment platform and Principal DevOps Architect and Manager in Europe.

Profile: www.rajeshkumar.xyz · Videos: youtube.com/@TheDevOpsSchool

20 years in industry	**200+** corporate trainings
10,000+ engineers mentored	**70+** organisations coached

REVIEWS

What learners say

Rated 4.6 on Google · more reviews at devopsschool.com/testimonials and in our video testimonials.

> “Rajesh is very good trainer. Rajesh was able to resolve our queries and question effectively. We really liked the hands-on examples covered during this training program.”
>
> Indrayani, India

> “The training was very useful and interactive. Rajesh helped develop the confidence of all.”
>
> Abhinav Gupta, Pune

START YOUR DEEP DIVE INTO DOCKER SECURITY TRAINING

Join the next batch

Weekday batches start on the 1st of every month and weekend batches on the first Saturday of every month, with max 10 learners each. 1:1 training can start immediately; corporate batches run on demand. We reply within 1 hour.

Pay by UPI, credit and debit cards, net banking or PayPal. Before you pay, read our refund policy. All levels of this course: www.devopsschool.com/courses/docker/

ENROL OR RESERVE A SEAT
Deep Dive into Docker Security

WHATSAPP · INDIA
+91 99057 40781

WHATSAPP · USA
+1 (469) 756-6329

EMAIL
contact@DevOpsSchool.com