DevOpsSchool

Course curriculum · November 2026

LEVEL 04 OF 04 · ADVANCED

# Kapitan Advanced Training

Run Kapitan as a platform for many teams with fast compiles, hardened secrets, policy checks and migrations.

| | | |
|---|---|---|
| LIVE ONLINE<br>**Max 8-10 hours** | CORPORATE (DAY-WISE)<br>**5 days · Days 1-5** | SELF-PACED VIDEO<br>**1-year LMS access** |
| BATCH SIZE<br>**Max 10 learners** | NEXT BATCHES<br>**1st of every month · weekends: first Saturday** | CERTIFICATE<br>**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 | 47 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/kapitan/kapitan-advance-training.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 Kapitan course path, you will be able to:

1

### Model configuration once

Describe every environment in a single Kapitan inventory and generate consistent manifests from it.

2

### Manage secrets safely

Keep secrets encrypted in Git with Kapitan refs and reveal them only at deploy time.

3

### Automate delivery with GitOps

Compile, validate and ship Kapitan output through CI pipelines and GitOps tools.

WHAT THE ADVANCED LEVEL ADDS

### Scale, security hardening and platform patterns

**Hands-on:** Migrate a service estate to Kapitan. Convert three services from Helm and Kustomize into Kapitan components, add policy checks and key rotation, and document the platform workflow.

WHO SHOULD ATTEND

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

BEFORE YOU START

- The Intermediate level, or equivalent hands-on experience.
- You can already: Automate Kapitan with generators, CI pipelines and GitOps delivery while keeping secrets out of Git.
- 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

# Kapitan Advanced Training agenda

47 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: Kapitan concepts, inventory and first compile

Included from Foundation

1. What Kapitan is: a configuration management tool that generates Kubernetes, Terraform, scripts and docs from one inventory
2. Why Kapitan: removing duplication across environments compared with Helm, Kustomize and plain Jsonnet
3. Installing Kapitan: pip install kapitan, the kapicorp/kapitan container image and kapitan --version
4. Project layout: inventory/targets, inventory/classes, components, Jinja2 sources and the compiled directory
5. Inventory basics: targets and classes in YAML, parameters and class inheritance
6. Parameter interpolation: ${...} references between parameters and target-specific variables
7. Inspecting the inventory: kapitan inventory -t and kapitan searchvar
8. First compile: kapitan compile, selecting targets with -t and reading the compiled output
9. Input types overview: jinja2, jsonnet, kadet (Python), helm, copy and external
10. Jinja2 input: rendering text files, scripts and README documentation from inventory values
11. Kapitan with Git: committing compiled output and reviewing diffs as the change record

## DAY 2 — Essential: Jsonnet, Kadet, Helm inputs and secrets

Included from Essential

12. Jsonnet input: writing Kubernetes manifests in Jsonnet and reading the inventory through kapitan.libsonnet
13. Kadet input: generating manifests in Python with BaseObj and kadet components
14. Helm input: rendering upstream Helm charts with values taken from the inventory
15. Output types: yaml, json and plain output, and setting output_path per component
16. Compile configuration: parameters.kapitan.compile entries with input_type, input_paths and output_type
17. External dependencies: fetching Git repositories, HTTP files and Helm charts with --fetch
18. Secrets with refs: the ?{...} reference syntax and kapitan refs --write and --reveal
19. Ref backends: plain, base64, gpg, gkms, awskms, azkms, vaultkv and vaulttransit
20. Generated secrets: random strings, RSA keys and base64 functions in ref definitions
21. Reusable classes: common, environment and component classes, overriding parameters per target
22. Validation: kapitan validate against Kubernetes schemas and kapitan lint for unused classes and refs
23. Team workflow: target-per-environment conventions, review of compiled diffs and pre-commit hooks

## DAY 3 — Intermediate: Generators, CI/CD and GitOps integration

Included from Intermediate

24. Kapitan generators: the kapicorp generators library for Kubernetes, Terraform and documentation
25. Kubernetes generator: defining deployments, services, config maps and secrets purely in the inventory
26. Terraform with Kapitan: compiling Terraform JSON per environment and configuring state backends
27. Scripts and docs: generating apply scripts and per-target README files with Jinja2
28. CI pipelines: kapitan compile --fetch in GitHub Actions or GitLab CI, failing on uncommitted diffs
29. GitOps delivery: syncing compiled manifests with Argo CD or Flux, and Kapitan as an Argo CD plugin
30. Reveal at deploy time: kapitan refs --reveal in pipelines while plaintext never reaches Git

31. Inventory backends: reclass, the faster reclass-rs and the OmegaConf backend
32. Kadet libraries: sharing Python component code across targets and unit testing it
33. Troubleshooting compiles: --verbose output, interpolation errors, missing classes and ref decryption failures
34. Access control: KMS key policies, Vault policies and per-environment secret isolation
35. Multi-cluster layout: organising targets by cluster, region and environment

DAYS 4-5 **Advanced: Scale, security hardening and platform patterns** — New at this level

36. Large inventories: compile speed with --parallelism, caching and the reclass-rs backend
37. Platform engineering: offering Kapitan-based golden paths to application teams
38. Security hardening: rotating KMS and GPG keys, re-encrypting refs and auditing reveal access
39. Policy checks: validating compiled output with kubeconform, Conftest (OPA) or the Kyverno CLI
40. Governance: class ownership, CODEOWNERS on inventory paths and change approval rules
41. Migration: moving existing Helm, Kustomize or raw YAML repositories into Kapitan components
42. Extending Kapitan: external input types and custom Jinja2 filters
43. Versioning and upgrades: pinning Kapitan versions and testing upgrades against compiled diffs
44. Multi-tool estates: one inventory driving Kubernetes, Terraform, Ansible variables and documentation
45. Disaster recovery: rebuilding environments from Git, secret backends and compiled output
46. Choosing an approach: Kapitan versus Helmfile, Kustomize overlays, Tanka or CUE
47. Capstone: design a multi-environment, multi-cluster Kapitan repository with secrets, CI and GitOps

PRACTICE

## Hands-on labs

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

FOUNDATION **Compile two environments from one inventory**
Create dev and prod targets that share classes, override a few parameters per target and compile Kubernetes manifests and a README for both.

ESSENTIAL **Deploy an application with Kapitan secrets**
Package an application as a Jsonnet or Kadet component, add a database password as a gkms or gpg ref and deploy the revealed manifests to a cluster.

INTERMEDIATE **Build a GitOps pipeline with Kapitan**
Create a CI job that compiles all targets, blocks merges with stale compiled output and lets Argo CD sync the result to two clusters.

ADVANCED **Migrate a service estate to Kapitan**
Convert three services from Helm and Kustomize into Kapitan components, add policy checks and key rotation, and document the platform workflow.

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 Kapitan 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 KAPITAN 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/kapitan/

ENROL OR RESERVE A SEAT
**Kapitan Advanced Training**

WHATSAPP · INDIA
**+91 99057 40781**

WHATSAPP · USA
**+1 (469) 756-6329**

EMAIL
**contact@DevOpsSchool.com**