+33 (0)1 48 10 09 46
formation@ambient-it.net
140 Avenue Jean Lolive 93500 Pantin - France

Updated on 06/05/2026

Sign up

Tetragon Training

3 days (21 hours)

Overview

Tetragon is an eBPF-based runtime security solution for Kubernetes, Linux, and cloud-native environments.

This security observability platform allows you to monitor processes, file access, network activity, and system calls directly at the kernel level.

Our Tetragon training will help you master threat detection and runtime enforcement by leveraging events enriched with Kubernetes context.

You will learn how to install Tetragon, configure its components, analyze process events, and create your own Tracing Policies to detect suspicious behavior.

By the end of the training, you will be able to deploy Tetragon in a Kubernetes cluster, export its events to your security tools, apply response rules, and build a cloud-native security observability strategy.

Like all our training courses, this one will introduce you to **the latest stable version** of the technology and its new features.

Objectives

- Understand how Tetragon works and its use with eBPF
- Install and configure Tetragon in a Kubernetes environment
- Analyze process, file, network, and syscall events
- Create tracing policies tailored to security use cases
- Implement runtime detection and enforcement rules
- Integrating Tetragon events into a SOC, SIEM, or observability pipeline

Target Audience

- Security and DevSecOps engineers
- Kubernetes administrators
- SRE and platform engineers
- Cloud-native SOC analysts
- Cloud and infrastructure architects

Prerequisites

- Basic knowledge of Kubernetes
- Basic Linux administration skills
- General understanding of cybersecurity and cloud-native environments
- Basic knowledge of containers and Docker is recommended

Technical prerequisites

- A computer running Linux, macOS, or Windows with WSL2
- Have the necessary installation permissions on your computer
- Install Docker or an equivalent container runtime
- Have a local or remote Kubernetes cluster, such as kind, minikube, or a managed cluster
- Install kubectl, Helm, and a code editor
- Ensure a stable internet connection

Tetragon Training

[Day 1 - Morning]

Understanding Tetragon, eBPF, and runtime security

- Understand Tetragon’s role in a runtime security strategy
- Identify the limitations of traditional approaches: application logs, userspace agents, SIEM alone
- Discover the benefits of eBPF for kernel observability and real-time detection
- Understanding monitored events: process_exec, syscalls, files, network, and privileges
- Deploying Tetragon in a Kubernetes, cloud-native, and DevSecOps environment
- Hands-on workshop: Install Tetragon on a test cluster and observe the first events

[Day 1 - Afternoon]

Architecture, installation, and initial events

- Understanding Tetragon’s architecture: agent, sensors, export, gRPC, and tetra CLI
- Installing Tetragon with Helm in a Kubernetes cluster
- Configure components, namespaces, DaemonSet, and basic settings
- Handling JSON events and compact or detailed output formats
- Identifying Kubernetes metadata: namespace, pod, container, labels, and node
- Diagnosing installation errors and validating the deployment’s proper functioning

Monitoring processes and system activities

- Analyze process lifecycle events with process_exec and process_exit
- Track commands executed in containers and on nodes
- Detect suspicious behavior: interactive shells, executions from /tmp, network tools
- Correlate process events with Kubernetes workloads
- Define an initial framework for identifying early warning signs of compromise
- Hands-on workshop: detecting abnormal executions in a Kubernetes pod

[Day 2 - Morning]

Tracing Policies and custom detection

- Understanding the structure of Tetragon Tracing Policies
- Define detection rules for kernel events, files, processes, and arguments
- Using in-kernel filtering to reduce noise and improve performance
- Building policies tailored to critical workloads
- Testing, versioning, and debugging detection policies
- Hands-on workshop: writing a Tracing Policy to detect access to a sensitive file

[Day 2 - Afternoon]

Cloud-native threat detection

- Identify common attack scenarios in Kubernetes: shell, escape, reconnaissance, exfiltration
- Detect changes in Linux privileges, capabilities, and namespaces
- Monitor suspicious network access and tools used during post-exploitation
- Correlate Tetragon events with MITRE ATT&CK tactics
- Prioritize alerts based on severity, Kubernetes context, and workload exposure
- Hands-on workshop: Simulate attack behavior and generate an actionable alert

Runtime enforcement and immediate responses

- Understanding Tetragon’s runtime enforcement capabilities
- Distinguish between observation, filtering, alerting, blocking, and response actions
- Configure policies to restrict certain executions or sensitive access
- Reduce the risk of TOCTOU by applying rules at the kernel level
- Defining a phased strategy between audit mode and enforcement mode
- Hands-on workshop: Blocking the execution of an unauthorized binary in a container

[Day 3 - Morning]

Export, observability, and SOC integration

- Understanding export formats: JSON, files, gRPC, and external integrations
- Send events to observability or centralization tools
- Structure events for use by a SIEM or SOC
- Building monitoring dashboards: processes, files, network, workloads, and alerts
- Reducing noise with filters, Kubernetes labels, and prioritization rules
- Hands-on workshop: Exporting Tetragon events to a security analytics pipeline

[Day 3 - Afternoon]

Performance, hardening, and operations

- Understanding the impact of policies on performance and event volume
- Apply best practices for filtering to limit overhead
- Monitor Tetragon status, metrics, logs, and runtime errors
- Securing the deployment: RBAC, namespaces, gRPC access, and role separation
- Implement a strategy for policy maintenance, upgrades, and validation
- Develop a production deployment checklist for a Kubernetes cluster

Governance, final case, and industrialization

- Define governance for detection and enforcement policies
- Integrate Tetragon into a DevSecOps and GitOps workflow
- Organize collaboration between security, SRE, platform, and SOC teams
- Document use cases: investigation, detection, blocking, and compliance
- Building a phased adoption roadmap across multiple clusters
- Hands-on workshop: create a comprehensive scenario for detection, alerting, enforcement, and reporting

To learn more

Target Audience

This training is intended for both individuals and companies, large or small, seeking to train their teams in new advanced IT technologies or to acquire specific business knowledge or modern methods.

Assessment upon enrollment

The pre-training assessment complies with Qualiopi quality standards. Upon final registration, the learner receives a self-assessment questionnaire that allows us to evaluate their estimated proficiency in various types of technologies, as well as their expectations and personal goals for the upcoming training, within the limits imposed by the selected format. This questionnaire also allows us to anticipate certain connection or internal security issues within the company (intra-company or virtual classroom) that could pose challenges for monitoring and ensuring the smooth running of the training session.

Teaching Methods

Practical Course: 60% Practical, 40% Theory. Training materials distributed in digital format to all participants.

Organization

The course alternates between theoretical input from the trainer, supported by examples and reflection sessions, and group work.

Assessment

At the end of the session, a multiple-choice questionnaire is used to verify that the skills have been properly acquired.

Certification

A certificate will be issued to each trainee who has completed the entire training program.