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

Updated on 09/02/2026

Sign Up

Certified Junior Detection Engineer Training and Certification

5 days (35 hours)

Overview

The Certified Junior Detection Engineer Certification Training prepares you for the official Certified Junior Detection Engineer (CJDE) certification and the fundamental threat detection skills expected in a modern SOC.

This intermediate-level course helps you build the knowledge needed to design, test, and improve actionable detections. You'll cover network fundamentals, Windows, Linux, log analysis, and incident response methods relevant to the CJDE exam.

You'll learn to create rules using Sigma and YARA, analyze network data with Zeek, and leverage the capabilities of SIEM platforms such as Splunk, Elastic, and Graylog. The goal is to develop a structured, contextualized, and measurable approach to detection.

The course also covers the integration of threat intelligence, alert tuning, behavioral analysis, the basics of Python, and Git practices applied to detection. The exercises are based on realistic scenarios to simulate situations encountered in an operational environment.

Upon completion of the training, you will be able to effectively prepare for the practical exam for the Certified Junior Detection Engineer (CJDE) certification. You will have the necessary skills to analyze a threat, develop a detection rule, reduce false positives, and present a consistent validation process.

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

Objectives

- Understand the areas of expertise assessed by the Certified Junior Detection Engineer (CJDE) certification.
- Analyze network, system, and application data to identify
.
- Create and validate detection rules using Sigma, YARA, and SIEM queries.
- Tune detections to improve their relevance and minimize false positives.
- Integrate threat intelligence and behavioral analysis into a detection process.
- Prepare for practical exercises and the CJDE exam format.

Target Audience

- SOC Analyst
- Detection Analyst
- Detection Engineer
- SIEM Administrator
- Incident Response Analyst

Prerequisites

- Fundamental knowledge of cybersecurity and how a SOC operates.
- Proficiency in the basics of TCP/IP, DNS, and HTTP networks.
- Basic experience with Windows and Linux environments.
- Ability to read technical logs and run simple queries.
- Basic scripting skills in Python or an equivalent language.

Technical Requirements

- Computer running Windows 10 or 11, macOS, or a recent Linux distribution.
- At least a dual-core processor; 8 GB of RAM recommended; and available disk space for work files.
- A recent version of Google Chrome, Mozilla Firefox, or Microsoft Edge, with the latest updates installed.
- Stable internet connection without security filtering that blocks lab platforms, downloads, and access to training services.
- Installation permissions or access to a work environment that allows the use of Visual Studio Code and Git.

Curriculum for our Certified Junior Detection Engineer training

[Day 1 - Morning]

Fundamentals of detection and overview of the certification

- Overview of the Certified Junior Detection Engineer (CJDE) certification and expected competencies.
- The role of detection engineering in a SOC, distinguishing it from threat hunting and incident response
  incident response.
- The detection lifecycle, from threat hypothesis to quality monitoring.
- Overview of telemetry sources, including endpoint, network, identity, and application logs.
- Review of TCP/IP, DNS, HTTP, routing, and network services concepts useful for analysis.
- Hands-on workshop: mapping log sources in a mock SOC and associating each source with the threats it can help detect.

[Day 1 – Afternoon]

System Analysis and Incident Response

- Essential security artifacts on Windows and Linux.
- Reading authentication, process, service, and system activity logs.
- Alert analysis workflow: qualification, enrichment, prioritization, and escalation.
- Indicators of compromise, suspicious behavior, and evidence.
- Reminders on containment, preservation of relevant evidence, and lessons learned.
- Hands-on workshop: Qualify a suspicious authentication alert based on Windows and Linux logs, then draft initial response actions.

[Day 2 - Morning]

Network and Security Telemetry

- Analysis of network traffic, sessions, protocols, and metadata.
- Using Wireshark and tcpdump to examine suspicious traffic.
- Identifying DNS anomalies, unusual outbound connections, and command-and-control communications.
- Understanding logs from firewalls, proxies, IDS, and network devices.
- Developing detection hypotheses based on observable attack techniques on the network.
- Hands-on workshop: Analyze a network capture to identify suspicious activity, extract relevant indicators, and define detection logic.

[Day 2 - Afternoon] Network

Detection with Zeek

- Zeek’s architecture and objectives within a network detection strategy.
- Reading key Zeek logs, including connection, DNS, HTTP, SSL, and file logs.
- Correlating network events to build an investigation scenario.
- Identifying behaviors associated with data exfiltration, scanning, beaconing, and suspicious downloads.
- Principles of creating, testing, and fine-tuning detection based on network telemetry.
- Hands-on workshop: Analyzing Zeek logs to detect a beaconing pattern and document detection tuning criteria.

[Day 3 - Morning]

Sigma Rules and Portable Detection

- Principles of declarative detection with Sigma and the structure of a rule.
- Selecting fields, operators, conditions, and metadata for a maintainable detection rule.
- Translating a threat scenario into event-based detection logic.
- Best practices for versioning, documenting, and validating a rule.
- Limitations of generic rules and adapting them to the organization’s data collection context.
- Hands-on workshop: Write a Sigma rule to detect suspicious execution, test it on provided events, and refine its conditions.

[Day 3 - Afternoon]

YARA and Malware-Oriented Analysis

- YARA’s objectives for identifying malware files and families.
- Rule syntax, strings, conditions, modifiers, and best practices for readability.
- Differences between static indicators, fragile signatures, and detectable behaviors.
- Leveraging file artifacts to enhance investigations and detection.
- Validating YARA rules, managing false positives, and documenting limitations.
- Hands-on workshop: Design a YARA rule based on harmless samples and provided artifacts, then evaluate its accuracy.

[Day 4 - Morning]

SIEM, Queries, and Alert Configuration

- Functions of a SIEM in centralizing, searching, and correlating events.
- Query approaches in Splunk, Elastic, and Graylog.
- Building a detection rule based on a query, a threshold, or a sequence of events.
- Measuring alert quality, false positive rates, coverage, severity, and actionable value.
- Tuning methods, justified exceptions, allowlists, and maintaining visibility.
- Hands-on workshop: Create a detection query in a SIEM, analyze the results, and fine-tune it to reduce irrelevant alerts.

[Day 4 - Afternoon]

Threat intelligence and behavioral detection

- Differences between strategic, operational, and tactical intelligence.
- Assessing the reliability, recency, and relevance of indicators of compromise.
- Integrating threat intelligence into searches, rules, and investigations.

- Behavioral approach, baselines, anomalies, and detections resilient to changes in indicators.
- Contributions and limitations of AI in detection and prioritization workflows.
- Hands-on workshop: enriching a compromise case with indicators and designing a complementary behavioral detection rule.

[Day 5 - Morning]

Detection as Code and Industrialization

- Principles of Detection as Code to make rules traceable, reusable, and controlled.
- Using Git, commits, branches, and code reviews applied to detection content.
- Organizing a rule repository, naming conventions, and metadata management.
- Testing strategy, validation datasets, and regression testing.
- Deployment preparation, post-deployment monitoring, and continuous improvement of detections.
- Hands-on workshop: versioning a Sigma rule in a Git repository, documenting its purpose, and preparing a test plan before deployment.

[Day 5 - Afternoon]

Review and preparation for the practical exam

- Comprehensive review of the topics covered by the Certified Junior Detection Engineer (CJDE) certification.
- Analysis of exam-style scenarios, evidence collection, prioritization, and technical justification.
- Time-management strategies for a long-duration practical assessment.
- Verification of deliverables, clarity of conclusions, traceability of decisions, and quality of documentation.
- Identification of individual review areas, common mistakes, and habits to adopt before the exam.
- Hands-on workshop: conducting a synthesis exercise combining log analysis, rule creation, configuration, and presentation of a detection case.

Target Companies

This training is intended for both individuals and companies—large or small—that wish to train their teams in new, advanced IT technology or to acquire specific industry knowledge or modern methods.

Entry-Level Assessment

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 level across 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 enables us to anticipate certain connection or internal security issues within the company (in-house or virtual classroom) that could

impede the monitoring and smooth running of the training session.

Teaching Methods

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

Organization

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

Assessment

At the end of the session, a multiple-choice quiz is administered to verify that participants have successfully acquired the skills.

Certification

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