AMBIENT-IT

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

Updated on 02/04/2026

Register

GoReleaser Training

2 days (14 hours)

Overview

GoReleaser automates the publication of your Go applications: multi-platform builds, packaging, and distribution in a single pipeline. Ideal for making your Git releases more reliable, speeding up delivery, and standardizing your artifacts (binaries, archives, images).

This training guides you through the industrialization of a reproducible release cycle: binary generation for Linux/macOS/Windows, version management, changelog, signatures, and publication on registries. You will learn how to structure a repository ready for continuous delivery, with a clear and maintainable configuration.

The approach is decidedly practical: .goreleaser.yaml configuration workshops, CI pipeline demos, and troubleshooting exercises (tags, permissions, checksum, cross-compilation). Deliverables include a sample project, a release checklist, and an operational pipeline ready to be adapted to your context.

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

Objectives

- Configure GoReleaser to generate multi-OS/CPU artifacts.
- Automate the creation of releases from Git tags.
- Produce changelogs, checksums, and signatures to secure distribution.
- Publish artifacts on a release platform and/or registry.
- Diagnose and correct common build and CI errors.

Target audience

- Go developers who want to industrialize delivery.
- DevOps/Platform engineers in charge of CI/CD pipelines.
- Tech leads responsible for release quality.

Prerequisites

- Good foundation in Go (modules, build, tests).
- Practical experience with Git (branches, tags, releases).
- Understanding of CI/CD and environment variables.
- Comfortable with the command line.

Technical prerequisites

- Minimum 8 GB RAM (16 GB recommended).
- Linux, macOS, or Windows (WSL2 recommended on Windows).
- Go installed (recent stable version) and access to the terminal.
- Git installed and a code editor.
- Docker optional for reproducible builds and packaging.

GoRealeaser training program

[Day 1 - Morning]

Getting started with GoReleaser and project preparation

- Understanding the role of GoReleaser: multi-platform builds, packaging, publishing
- Project prerequisites: Go module, package structure, reproducible go build command
- Versioning management: Git tags, SemVer conventions, release notes
- Initializing the configuration: .goreleaser.yaml, init and check commands
- Hands-on workshop: Tagging a v0.1.0 and generating a first local build with GoReleaser.

[Day 1 - Afternoon]

Builds, archives, and artifacts: mastering configuration

- Configuring builds: GOOS/GOARCH, CGO, flags, ldflags (version, commit, date)
- Generating archives: formats, naming, file inclusion (LICENSE, README, supplements)
- Checksums and signatures: sha256, best practices for integrity
- Changelog: automatic generation, filters, grouping by commit type
- Hands-on workshop: Producing Linux/Windows/macOS binaries with standard naming and checksums.

[Day 2 - Morning]

Release publishing and CI/CD automation

- Publishing on GitHub/GitLab: releases, assets, permissions, and tokens
- Execute in CI: GitHub Actions workflow (checkout, setup-go, cache, goreleaser)
- Secret management: GITHUB_TOKEN, personal tokens, minimal scopes
- Publishing strategies: dry-run, snapshots, pre-releases, branches, and tags
- Hands-on workshop: Set up a CI pipeline that automatically publishes to each tag.

[Day 2 - Afternoon]

Advanced packaging: Homebrew, Docker, and distribution

- Generate a Homebrew formula: tap, dependencies, tests, and conventions
- Build and publish Docker images: tags, multi-arch, labels, SBOM if required
- OS packages: deb/rpm (NFPM), pre/post install scripts, config files
- Ensuring reliable delivery: reproducibility, matrices, artifact validation, rollback
- Hands-on workshop: Publish a release including a multi-arch Docker image and a Homebrew formula.

Target companies

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

Positioning at the start of training

The positioning at the start of the training complies with Qualiopi quality criteria. Upon final registration, the learner receives a self-assessment questionnaire that allows us to assess their estimated level of knowledge of different types of technologies, their expectations and personal objectives 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 be problematic for the monitoring and smooth running of the training session.

Teaching methods

Practical training: 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 discussion sessions, and group work.

Assessment

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

Certification

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