

GitOps training

2 days (14 hours)

Presentation

GitOps is an approach to application development and operation based on Git, a widely used version control system. It is a method for managing application infrastructure and configurations, using Git as a single source of truth and automating deployment processes.

This tool is an operational model for cloud-native technologies such as Kubernetes. GitOps provides a set of practices for deploying, managing and controlling clusters and containerized applications. The secret of GitOps' success lies in application management for developers, and the configuration of end-to-end CI/CD pipelines and Git workflows by operations and development teams.

Your teams become more productive with GitOps by making better use of Git tools to manage Kubernetes features faster. GitOps also integrates agility principles to respond quickly and appropriately to market needs.

Deployment and infrastructure management processes are automated as far as possible, using automation tools such as continuous integration/continuous deployment (CI/CD) pipelines to ensure consistency and reproducibility of deployments. Adopting this model will also enhance your security, thanks to the powerful cryptography provided by Git. Should a security breach occur, you'll be able to quickly recreate a new system independently of the compromised one.

GitOps focuses on visibility and observability of infrastructure and applications. Monitoring and logging tools are used to track the state of the system and detect deviations from the state declared in Git repositories.

With our GitOps training, your applications will be more compliant and secure thanks to a log audit listing all the changes applied to the cluster, as well as an audit trail indicating the actions carried out in your cluster with the aim of ensuring compliance.

comply with SOC 2 standards and ensure its stability.

Objectives

- Know and apply GitOps best practices
- Improve - Automate application deployment processes
- Understand the principles and issues behind CI / CD concepts
- Securing and recovering deployments in the event of failure
- Synchronize and automate applications in clusters with Git

Target audience

DevOps engineers, architects, system administrators.

Prerequisites

- Ideally have taken our Kubernetes training course, or have basic knowledge of containers
- Basic knowledge of Git

GitOps training program

GitOps fundamentals

- Introducing GitOps
- Why use GitOps?
- GitOps best practices

The various GitOps tools

- GitLab
- Github
- Jenkins
- Argo CD
- Flow
- Terraform

The GitOps deployment process

- Pipelines and the GitOps architecture
- Development on Git
- Declarative infrastructure and infrastructure as code (IaC)

- Setting up the CI/CD tool chain
- The different Continuous Deployment strategies
- Deploying a group of containerized applications
- Deploying a cluster in the cloud
- Managing scalability
- Automated deployment
- Manage and monitor cluster applications with ArgoCD

Optimizing the use of GitOps

- Version management
- Continuous integration
- Continuous delivery or continuous deployment
- Automate application creation and deployment

Best practices on Git

- The Issues
- Git Flow
- Quotas and Git cloud pricing
- Hosting your Git server

Companies concerned

This course is aimed at both individuals and companies, large or small, wishing to train their teams in a new advanced computer technology, or to acquire specific business knowledge or modern methods.

Positioning on entry to training

Positioning at the start of training complies with Qualiopi quality criteria. As soon as registration is finalized, the learner receives a self-assessment questionnaire which enables us to assess his or her estimated level of proficiency in different types of technology, as well as his or her expectations and personal objectives for the training to come, within the limits imposed by the selected format. This questionnaire also enables us to anticipate any connection or security difficulties within the company (intra-company or virtual classroom) which could be problematic for the follow-up and smooth running of the training session.

Teaching methods

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

Organization

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

Validation

At the end of the session, a multiple-choice questionnaire verifies the correct acquisition of skills.

Sanction

A certificate will be issued to each trainee who completes the course.