

Updated on 12/17/2024

Sign up

# NixOS training

2 days (14 hours)

## Presentation

The [NixOS](#) training course will introduce you to this Linux distribution based on the cross-platform Nix **package manager**. You'll discover its different operating paradigm thanks to its reproducibility and declarativeness, as well as a first approach to Nix and portable development environments.

During this training course, you'll be taught how to install, configure and upgrade NixOS, so that you're ready to use these solutions without encountering any major problems.

Become able to reproduce a complete system on another machine, or control dependencies to minimize the volume of applications. Learn the Nix language for describing packages, configurations, variants and compositions.

This course will be presented with the latest version of NixOS: [NixOS 24.05 \(Uakari\)](#).

## Objectives

- Developing an ISO
- Controlling different types of packages
- Understanding the benefits of NixOS: Reproducibility and declarability
- Writing a NixOS module
- Know the reproducibility of a system

## Target audience

- Linux system administrators

- Developers

## Prerequisites

General Linux knowledge.

## Software requirements

- Virtualization capabilities (e.g. Virtualbox) with 4GB for the guest
- Link to [ISO](#) Nixos
- A GitHub account may be useful, but is not necessary.

## NixOS training program

### Introduction and basic concepts

- Classic package operation on Linux
  - Package and package manager on Linux
  - Dependency and conflict management
  - Limitations (reproducibility, imperialism, multiple versions...)
- Introduction to NixOS
  - NixOS history
  - How it works and how it differs from other Linux distributions
  - Package management and reproducible development environment
- Nix basic concepts
  - Introducing the Nix package management language
  - Basic concepts: Store Nix, derivations, Nix expressions
- Installing NixOS
  - Installing NixOS on a virtual machine
  - Define system configuration via `/etc/nixos/configuration.nix`
  - Creation of an initial configuration with basic packages and services (e.g. SSH, Git, Vim)
  - Building and activating the configuration

### Package management with Nix

- Nix and the Nix language
  - Nix-env principle for managing non-NixOS user environments
  - Nix repl and nixpkgs
- Reproducible development environments
  - The `shell.nix` and `default.nix` files
  - Create project-specific development environments (e.g. Python projects)

### Advanced NixOS configuration

- System configuration with NixOS
  - Breakdown of /etc/nixos/configuration.nix configuration file
  - Adding services and managing users
  - Setting up system services (e.g. Nginx, Docker, Dolibarr)
- Understanding overlays and advanced options
  - Overlays for extending and customizing packages
  - Using NixOS modules
  - Create a modular configuration by separating Nix files
- NixOS maintenance
- Version management and rollback
  - System configuration update and version management
  - Revert to a previous configuration with immutable history
  - Cleaning the NixStore
  - NixOS channels
- Some practical features
  - Create clone VMs for testing in a single command
  - Different configuration switch methods
  - search.nixos
  - NixOps

## 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.