

## Docker

### Course Summary

#### Description

Linux containers are changing the way companies think about service development and deployment. Containers play a vital role in the modern data-center, and Docker is leading the way. This course covers all the core features of Docker including: container creation and management, interacting with Docker hub, using Dockerfile to create and manage custom images, advanced Docker networking (how to safely expose container services to the world, and link containers), the use of Docker volumes to manage persistent data, and Docker Compose to build multi-container applications. Emphasis is placed on best practices and how to secure Docker installations and containers. The course culminates with comprehensive labs where students use Docker, Git, and a continuous integration server to automate the testing of containerized applications.

Many large companies are moving an increasing number of applications to run inside containers. Containers can provide a high level of security while also making maximal use of hardware allowing higher density than traditional VMs. Docker is a dominant force in Linux containers and the core building block for nearly all higher level container management systems. This training will get sysadmins proficient with Docker so that they can immediately be productive as organizations continue to move applications onto containers.

#### Topics

- Container Technology Overview
- Installing Docker
- Managing Containers
- Managing Images
- Creating Images with Dockerfile
- ENV and WORKDIR
- Docker Networking
- Docker Volumes
- Docker Compose/Swarm
- Docker Registry Lab Tasks
- Continuous Integration with Gitlab, Gitlab Ci, And Docker Lab Tasks

#### Audience

This training will get sysadmins proficient with Docker so that they can immediately be productive as organizations continue to move applications onto containers.

#### Prerequisites

Before taking this course, you should have proficiency with the Linux CLI and a broad understanding of Linux system administration.

#### Duration

Three days

## Docker

### Course Outline

- I. Container Technology Overview**
  - A. Application Management Landscape
  - B. Application Isolation
  - C. Resource Measurement and Control
  - D. Container Security
  - E. Container Security
  - F. Open Container Initiative
  - G. Docker Ecosystem
- II. Lab Tasks**
  - A. Container Concepts runC
  - B. Container Concepts Systemd
- III. Installing Docker**
  - A. Installing Docker
  - B. Docker Architecture
  - C. Starting the Docker Daemon
  - D. Docker Daemon Configuration
  - E. Docker Control Socket
  - F. Enabling TLS for Docker
- IV. Lab Tasks**
  - A. Installing Docker
  - B. Install Docker via Docker Machine
  - C. Protecting Docker with TLS
- V. Managing Containers**
  - A. Creating a New Container
  - B. Listing Containers
  - C. Viewing Container Operational Details
  - D. Running Commands in an Existing Container
  - E. Interacting with a Running Container
  - F. Stopping, Starting, and Removing Containers
  - G. Copying files in/out of Containers
  - H. Inspecting and Updating Containers
- VI. Lab Tasks**
  - A. Managing Containers
  - B. Configure a Docker container to start at boot.
- VII. Managing Images**
  - A. Docker Images
  - B. Listing and Removing Images
- VIII. Lab Tasks**
  - A. Docker Images
  - B. Docker Platform Images
- IX. Creating Images With Dockerfile**
  - A. Dockerfile
  - B. Caching
  - C. docker build
  - D. Dockerfile Instructions
- X. ENV and WORKDIR**
  - A. Running Commands
  - B. Getting Files into the Image
  - C. Defining Container Executable
  - D. Best Practices
- XI. Lab Tasks**
  - A. Dockerfile Fundamentals
- XII. Docker Networking**
  - A. Overview
  - B. Data-Link Layer Details
  - C. Network Layer Details
  - D. Hostnames and DNS
  - E. Local Host <--> Container
  - F. Container <--> Container (same node)
  - G. Container <--> Container: Links
  - H. Container <--> Container: Private Network
  - I. Managing Private Networks
  - J. Remote Host <--> Container
  - K. Multi-host Networks with Overlay Driver
- XIII. Lab Tasks**
  - A. Docker Networking
  - B. Docker Ports and Links
  - C. Multi-host Networks

## Docker

### Course Outline (cont'd)

#### XIV. Docker Volumes

- A. Volume Concepts
- B. Creating and Using Volumes
- C. Managing Volumes (cont.)
- D. Changing Data in Volumes
- E. Removing Volumes
- F. Backing up Volumes
- G. SELinux Considerations
- H. Mapping Devices

#### XV. Lab Tasks

- A. Docker Volumes

#### XVI. Docker Compose/Swarm

- A. Concepts
- B. Compose CLI
- C. Defining a Service Set
- D. Docker Swarm (Legacy)

#### XVII. Lab Tasks

- A. Docker Compose
- B. Docker Swarm

#### XVIII. Docker Registry Lab Tasks

- A. Docker Registry
- B. Docker Registry (secured)

#### XIX. Continuous Integration With Gitlab, Gitlab Ci, And Docker Lab Tasks

- A. GitLab and GitLab CI Setup
- B. Unit and Functional Tests