

Linux Containers and Virtualization for Wind River

Length: 3 Days

Course Overview: The Linux Containers and Virtualization course provides engineers with a fast, cost-effective way to become familiar with, and deploy, Linux virtualization solutions based on containers or virtual machines.

After this course, participants will be able to perform the following:

- Understand technologies present in the Linux virtualization field and their internals
- Configure, build, and optimize an OpenEmbedded/Yocto Project/Wind River® Linux based host system that can accommodate virtualized guests
- Design, implement, and deploy various Linux containers or virtual machines with guest systems

Audience:

- Developers responsible for evaluation of Linux virtualization technologies
- Developers responsible for design, implementation, and deployment of Linux virtualization projects

Course Objectives:

- Utilize the Microsoft BI stack
- Implement scorecards, dashboards, and KPIs

Prerequisite Skills:

- Functional knowledge of Linux
- Basic understanding of operating systems and debugging techniques

COURSE CONTENT

Day 1: INTRODUCTION TO VIRTUALIZATION

- What is virtualization?
- Virtual machines
- Containers
- Implementation and deployment
- LAB: Configuring and Building a Virtualized System

HARDWARE VIRTUALIZATION

- Overview
- CPU
- Memory
- Networking

QEMU AND KVM

- Overview
- Usage
- Networking
- Debugging

LINUX CONTAINERS (LXC)

- Linux containers overview
- Namespaces
- Cgroups
- Linux containers configuration and workflow
- LAB: Working with Linux Containers
- LAB: Networking with Linux Containers
- LAB: Working with the Linux Containers Web Panel

YOCTO PROJECT VIRTUALIZATION

- Yocto Project overview
- Meta-virtualization layer

OVERC

- OverC overview
- OverC architecture
- OverC commands
- OverC workflows
- LAB: Working with OverC Containers

DOCKER

- Docker overview
- Working with Docker
- LAB: Working with Docker Containers
- LAB: Networking Docker Containers
- LAB: Creating Docker Images

OPEN CONTAINER INITIATIVE

- Open Container Initiative overview
- Open Container Initiative runtime specification
- Open Container Initiative image format specification
- Open Container Initiative image distribution specification

Day 2: RUNTIME ENVIRONMENTS

- Kvmtool
- Libvirt and Virsh
- Runv and Kata containers
- Runc
- Containerd
- Container runtime interface

REAL-TIME CONSIDERATIONS

- Planning a hypervisor
- Benchmarking virtualization platforms
- Debugging virtualization platforms
- Tuning virtualization platforms

STORAGE

- OpenStack storage
- QEMU KVM storage

Day 3: NETWORK VIRTUALIZATION

- Network virtualization essentials
- QEMU networking
- LXC networking

SECURITY

- Container security overview
- Container security mechanisms

ORCHESTRATION AND MONITORING

- Introduction to container orchestration
- Container orchestrator examples
- Container orchestration within OpenStack
- LAB: Working with the Docker Swarm Orchestrator

RESTFUL APIS

- Introduction to REST API calls
- Using RESTful APIs with containers