

Linux Kernel Modules (LFWS308)

OEM: Linux Foundation • Duration: 1 Day (8 hrs) • Code: LFWS308

COURSE MODULES & TOPICS

Introduction

- Objectives
- Who You Are
- The Linux Foundation
- Copyright and No Confidential Information
- The Linux Foundation Training
- Certification Programs and Digital Badging
- The Linux Foundation THRIVE-ONE Subscriptions
- Linux Distributions
- Platforms
- Preparing Your System
- Using and Downloading a Virtual Machine
- Things Change in Linux and Open Source Projects
- Documentation and Links

Kernel Modules Introduction

- What does an operating system do?
- Aside: Additional Rings of Protection
- Virtual Memory
- Conceptual View of Page Tables
- Process Context Switching
- Performance Cost of Virtual Memory
- Virtual Memory Layout
- Linux Memory Map: 32-bit
- Linux Memory Map: 64-bit
- Process-to-Process Context Switches
- User mode/space restrictions
- How Does User Mode Interact with Kernel Mode?
- Summary
- Labs

How to Work in OSS Projects

- Overview on How to Contribute Properly
- Know Where the Code is Coming From: DCO and CLA
- Stay Close to Mainline for Security and Quality
- Study and Understand the Project DNA
- Figure Out What Itch You Want to Scratch
- Identify Maintainers and Their Work Flows and Methods
- Get Early Input and Work in the Open
- Contribute Incremental Bits, Not Large Code Dumps
- Leave Your Ego at the Door: Don't Be Thin-Skinned
- Be Patient, Develop Long Term Relationships, Be Helpful

Kernel Modules Overview

- What is a Kernel Module?
- What are Kernel Modules Used For?
- Alternatives to Kernel Modules
- Security Concerns
- Kernel Module Signing
- Kernel Module Signing
- Are Signed Kernel Modules Required?
- Tainted Kernel
- Summary
- Labs

Building Kernel Modules

- How a Kernel Module is Built
- Installing Kernel Header Files
- Verify the Installation
- Structure of a Kernel Module
- Sample Code
- Compiling a kernel module
- How to install a kernel module
- Confirmation of module operation
- Tainted kernels
- Kernel module parameters
- Sample Parameter Code
- Working with parameters
- Summary
- Labs

Interaction with User Space

- User-to-Kernel Interactions
- Challenges of Kernel-to-User Interaction
- Comparing Interfaces
- procfs
- sysfs
- Module Parameters
- ioctl
- struct file_operations
- Typical Kernel Module Interactions
- Summary
- Labs

Closing and Evaluation Survey

- Evaluation Survey